PB82-226440

Application of Cluster Analysis to Aerometric Data. Volume II. Part 3 Modifications and Options Applied to Wolfe's NORMIX 360 Cluster Analysis Program

(U.S.) Environmental Monitoring Systems Lab.
Research Triangle Park, NC

Dec 80

U.S. Department of Commerce
National Technical Information Service
NTIS.

PB82-226440

EPA-600/4-80-048B
December 1980

APPLICATION OF CLUSTER ANALYSIS
TO
AEROMETRIC DATA

Volume II

Edited by

Raymond C. Rhodes
Quality Assurance Specialist

Part 3: Modifications and Options Applied to Wolfe's
NORMIX 360 Cluster Analysis Program
by Maurice E. Graves
Northrop Services, Inc.

December 1980

ENVIRONMENTAL MONITORING SYSTEMS LABORATORY
OFFICE OF RESEARCH AND DEVELOPMENT
U.S. ENVIRONMENTAL PROTECTION AGENCY
RESEARCH TRIANGLE PARK, NORTH CAROLINA 27711

REPRODUCED BY
NATIONAL TECHNICAL
INFORMATION SERVICE
U.S. DEPARTMENT OF COMMERCE
SPRINGFIELD, VA 22161

TECHNICAL REPORT DATA
(Please read Instructions on the reverse before completing)

1. REPORT NO. EPA-600/4-80-048-b

2.

3. RECIPIENT'S ACCESSION NO. PB82 226440

4. TITLE AND SUBTITLE Application of Cluster Analysis to Aerometric Data, Volume II. Part 3: Modifications and Options Applied to Wolfe's NORMIX 360 Cluster Analysis Program

5. REPORT DATE December 1980

6. PERFORMING ORGANIZATION CODE

7. AUTHOR(S)
Editor: Raymond C. Rhodes
Part 3: Maurice E. Graves, Northrop Services Inc.

8. PERFORMING ORGANIZATION REPORT NO.

9. PERFORMING ORGANIZATION NAME AND ADDRESS
U.S. Environmental Protection Agency
Environmental Monitoring Systems Laboratory
Research Triangle Park, NC 27711

10. PROGRAM ELEMENT NO.

11. CONTRACT/GRANT NO.

12. SPONSORING AGENCY NAME AND ADDRESS

13. TYPE OF REPORT AND PERIOD COVERED

14. SPONSORING AGENCY CODE EPA 600/08

15. SUPPLEMENTARY NOTES

16. ABSTRACT

The calibration and enhancement of Wolfe's NORMIX (normal mixtures) computer program in the National Computing Center of the U.S. Environmental Protection Agency at the Research Triangle Park, NC is documented. The program is available for data clustering, validation, and classification.

In Part 1, an historical perspective of the NORMIX algorithm is presented with details of calibration, using the Fisher-Anderson Iris data set. New program options are then introduced with illustrative outputs including clustering features such as variate profiles and bivariate representations.

In Part 2, SAS and MIKCA clustering algorithms are applied to LACS data to investigate their performance as a multivariate tool for modeling air pollution data.

In Part 3, a discussion of new NORMIX modifications and options is provided with complete program listings and other user aids.

In Part 4, tabular and graphical displays record some aspects of the separation of environmenal data into clusters by the NORMIX program. Three CHAMP monitoring sites are principal data sources.

This volume, Volume II, of a 3-volume series comprises the report for Part 3. Parts 1 and 2 are reported in Volume I; Part 4 is reported in Volume III.

17. KEY WORDS AND DOCUMENT ANALYSIS

a. DESCRIPTORS	b. IDENTIFIERS/OPEN ENDED TERMS	c. COSATI Field/Group
Cluster Analysis Data Validation Data Classification Outliers Statistics Environmental Data Computer Program (NORMIX)	Environmental Monitoring Data Management	43F 68A

18. DISTRIBUTION STATEMENT RELEASE TO PUBLIC	**19. SECURITY CLASS** *(This Report)* UNCLASSIFIED	**21. NO. OF PAGES** 188
	20. SECURITY CLASS *(This page)* UNCLASSIFIED	**22. PRICE**

EPA Form 2220-1 (Rev. 4-77) PREVIOUS EDITION IS OBSOLETE

DISCLAIMER

This report has been reviewed by the Environmental Monitoring Systems Laboratory, U.S Environmental Protection Agency, and approved for publication. Mention of trade names or commercial products does not constitute endorsement or recommendation for use.

FOREWORD

This report represents the results of work performed by Northrop Services, Inc., under Contract Number 68-02-2566 for the Performance Evaluation Branch, Environmental Monitoring and Support Laboratory, U.S. Environmental Protection Agency, Research Triangle Park, North Carolina. This work was conducted in response to Technical Directive 2.3-3.

ABSTRACT

The NORMIX cluster analysis computer program has a 15-year history of improvements in technique and adaptations to different computer systems. During the recent adaptation to the UNIVAC 1110 system, the techniques and options of the 1977 version of the Wolfe NORMIX program have been preserved for the most part. However, a number of changes were made and several options added. They are covered in detail in this report.

A summary of changes and additions within the NORMIX program is presented first, followed by discussion of the substantial number of options available to a user of the NORMIX program. The appendices provide listings of the routines and subroutines, as well as user aids.

CONTENTS

Page

Disclaimer ii

Foreword iii

Abstract iv

Figures. x

Tables xi

Acknowledgments. xii

1.0 Introduction 1-1

2.0 1979 Modifications to the NORMIX Program 2-1

2.1 Background 2-1

2.2 Summary of Changes and Additions Within the NORMIX Program 2-1

2.2.1 MAIN 2-2

2.2.2 AITKEN and Function Area 2-3

2.2.3 ASSIGN 2-3

2.2.4 CHISQP 2-5

2.2.5 DENSIT 2-5

2.2.6 DUMPER 2-5

2.2.7 EIGEN. 2-5

2.2.8 HISTOG 2-6

CONTENTS (continued)

Page

2.2.9 INFORM 2-6

2.2.10 INITLE 2-7

2.2.11 KMEAN 2-7

2.2.12 MAPPER 2-8

2.2.13 MOMENT 2-8

2.2.14 NORMIX 2-8

2.2.15 NROOT 2-9

2.2.16 OUTPUT 2-9

2.2.17 PRTTYP 2-10

2.2.18 RELIST 2-10

2.2.19 REVALU 2-10

2.2.20 SYMINV 2-11

2.2.21 TYTLE 2-11

2.2.22 UPLOW 2-11

2.2.23 PUTMSG 2-11

2.2.24 NORM 2-11

2.2.25 TABLE 2-11

2.2.26 PROFILE 2-12

2.2.27 SEED 2-12

2.2.28 SEED1 2-12

CONTENTS (continued)

Page

3.0 Options in NORMIX Program and NORMIX Flow 3-1

3.1 NORMIX-PREP Options 3-3

3.1.1 Chi-Square Tests 3-3

3.1.2 Histograms (IHIST) 3-3

3.1.3 Probability Charts (IPROB) 3-4

3.1.4 Pseudo-Random Numbers (IRAND, FACTOR, IQOL) 3-4

3.1.5 Scaling (THRESH, MINPOL) 3-4

3.1.6 Wind Components (NWIND, IWIND) 3-5

3.2 NORMIX Options for Grouping Phase 3-5

3.2.1 Size of Subsample (MAXAV) 3-6

3.2.2 Grouping Pattern (IHIER) 3-6

3.2.3 Merge Details (MERGE) 3-6

3.2.4 Eigenanalysis Printout (IGEN) 3-6

3.3 NORMIX Options for Clustering Phase 3-7

3.3.1 Chi-Square Tests and Histograms (IHIS) 3-7

3.3.2 Minimum Number of Points in Cluster (ENMIN) 3-7

3.3.3 Number of Hypothesized Types (NTYP) 3-7

3.3.4 Probability Level for Chi-Square Test (HTEST) 3-7

3.3.5 Override Chi-Square Test Abort (IOVER) 3-8

3.3.6 Printing at Each Iteration (IPRINT) 3-8

3.3.7 Time Limit on Computations (LTIME) 3-8

CONTENTS (continued)

Page

3.3.8 Maximum Number of Iterations (ITERX) 3-9

3.3.9 Correlation Matrices: Same or Different (NOR) 3-9

3.3.10 Time Printouts (MTIME) 3-9

3.3.11 Backtransformation Statistics (IBACK, NWIND, IWIND) 3-9

3.3.12 Input of Preliminary Estimates 3-10

3.4 CALCOMP Options 3-11

4.0 Discussion of NORMIX Program and Subprograms 4-1

4.1 NORMIX-PREP 4-1

4.1.1 MAIN 4-1

4.1.2 CUMPLT 4-2

4.1.3 NORM 4-3

4.1.4 SEED 4-3

4.2 NORMIX 4-3

4.2.1 NORMIX-PREP 4-3

4.2.2 INFORM, INITLE 4-5

4.2.3 HISTOG 4-5

4.2.4 REVALU 4-7

4.2.5 AITKEN 4-8

4.2.6 OUTPUT 4-9

4.2.7 ASSIGN 4-10

4.3 Cluster Variate Profiles 4-11

CONTENTS (continued)

Page

4.4 Probability Ellipses 4-13

4.5 Backtransformation Statistics 4-14

4.6 Parameter Listings and Other User Aids 4-14

4.7 Recommendations for Future Work 4-15

5.0 References 5-1

6.0 Appendices

6.1 Appendix A. Listing of NORMIX Subroutines 6-1

6.2 Appendix B. Listing of NORMIX-PREP Subroutines 6-69

6.3 Appendix C. Listing of Profile Plotting Routine. 6-83

6.4 Appendix D. Listing of Ellipse Plotting Routine. 6-89

6.5 Appendix E. Parameters and User Aids for NORMIX and Associated Routines. 6-107

6.6 Appendix F. Illustrative Outputs 6-121

FIGURES

Number | Page

1 General flow of NORMIX program and its associated routines with options listed underneath. 3-2

2 Detailed flow of NORMIX subroutines and associated routines . . 4-4

3 Cluster variate profile plot for three clusters with eight reordered variates. 4-12

E-1 Cluster variate profile plot for three clusters with eight reordered variates. 6-114

E-2 CALCOMP ellipse plot for two clusters and eight variates. . . . 6-117

F-1 Cluster variate profile plot for three clusters with eight reordered variates. 6-137

FIGURES (continued)

Number Page

F-2 CALCOMP ellipse plot for two clusters and eight variates. 6-139

TABLES

Number Page

1 NORMIX History, 1965-1979 1-2

E-1 NORMIX Parameters 6-108

E-2 NORMIX-PREP Parameters. 6-111

ACKNOWLEDGMENTS

The NORMIX computer program was developed by Dr. John H. Wolfe of the U.S. Navy, and was modified by Dr. Harold L. Crutcher and Mr. Raymond L. Joiner of the National Climatic Center. Mr. Raymond C. Rhodes of the U.S. Environmental Protection Agency acted as Project Director and Dr. Harold L. Crutcher as Consulting Scientist during the recent adaptation to the UNIVAC 1110 system. Appreciation is expressed to the U.S. Navy and to Dr. Wolfe for permission to adapt extensively the NORMIX program. Programming assistance was provided by Mr. James A. Stephenson of Integrated Systems, Inc., who wrote the NORMIX-PREP and ELLIPSE routines, and by Ms. Carole B. Cameron of Northrop Services, Inc., who refined the ellipse-plotting capability and developed the profile-drawing feature.

1.0 INTRODUCTION

The cluster analysis computer program was originated by Dr. John H. Wolfe (1965, 1967, 1971) in 1965, and called NORMIX to signify a "normal mixture analysis." Over the next 15 years the techniques were improved upon and adaptations to different computer systems developed. This history is tabulated in abbreviated form in Table 1. The program was recently adapted to the UNIVAC 1110 system for the U.S. Environmental Protection Agency (EPA) and has incorporated various options.

TABLE 1. NORMIX HISTORY, 1965 - 1979

Year	Researcher	Location	Feature	Computer
1965	John H. Wolfe	Naval Personnel and Training Research Laboratory, San Diego, CA	Maximum likelihood analysis of types by Newton-Raphson iteration; probability of membership in clusters	CDC 1604
1967	John H. Wolfe	Naval Personnel and Training Research Laboratory, San Diego, CA	Aitken iteration	CDC 1604
1969	John H. Wolfe	Naval Personnel and Training Research Laboratory, San Diego, CA	Option of same covariance matrix for each type; dynamic storage allocations	IBM 360
1971	John H. Wolfe	Naval Personnel and Training Research Laboratory, San Diego, Ca	KMEANS analysis and iterative hierarchical grouping	IBM 360
1972- 1977	Harold L. Crutcher, Raymond L. Joiner	National Climatic Center, Asheville, NC	Discriminant analysis for probability of membership; improved matrix handling	RCA Spectra 70
1979	Harold L. Crutcher,	Certified Consulting Meteorologist, 35 Westall Ave., Asheville, NC 28804	Double precision; display of cluster distributions; profile plots of scaled cluster statistics; supplementary program to transform input data, convert wind data, display input and transformed data, and add a scaled, normal random number; supplementary program to plot bivariate data and their probability ellipses for variable pairings.	UNIVAC 1110
	Raymond C. Rhodes,	EPA, Research Triangle Park, NC 27711		
	Maurice E. Graves,	Northrop Services, Inc., Research Triangle Park, NC 27709		
	James A. Stephenson	Integrated Systems, Inc., Research Triangle Park, NC 27709		

2.0 1979 MODIFICATIONS TO THE NORMIX PROGRAM

2.1 BACKGROUND

The techniques and options of the 1977 version of the Wolfe NORMIX program were basically preserved in the adaption to the UNIVAC 1110 system. The changes and additional options occurring in the NORMIX program are described in this section, first from an overall perspective, and then in detail by subroutine. A particular change can be pinpointed by observing the righthand side line numbers of the 1977 version which have remained intact, whereas new or changed lines are not numbered in the current version. Many of the options exist in the three associated programs called NORMIX-PREP, PROFILE, and ELLIPSE; they are discussed in subsequent sections. The NORMIX program that is being executed on the UNIVAC 1110 at EPA, Research Triangle Park, North Carolina, is given in Appendix A.

2.2 SUMMARY OF CHANGES AND ADDITIONS WITHIN THE NORMIX PROGRAM

Prior to 1979, double precision was used in the NORMIX program only to a limited extent. Double precision was mandatory in matrix inversions and available in the eigenanalysis subroutines, EIGEN and NROOT. Elsewhere, single precision was used for input data array and certain other variables. The new version is entirely in double precision except when calling International

Mathematical and Statistical Libraries (IMSL) subroutines, which are in single precision.

An important addition is present in Subroutine ASSIGN, where IMSL subroutines, GFIT and USHIST, are called to perform a chi-square test for normality and a frequency histogram display for each cluster that has a sufficient number of points, i.e., 20.

A second important addition is Subroutine PROFILE, whose plots of scaled parameter values and computations are prepared for a multicolored graphics display of cluster statistics. This subroutine is called from Subroutine OUTPUT.

The following subroutines constitute the present NORMIX program. They are listed with details on newly instituted features. The line numbers refer to the new version.

2.2.1 MAIN

a. Most of subsection 6. (Input) under C. (Usage): deleted or transferred to INFORM. The input parameters are now read by a NAMELIST.

b. Subsection 8. (Assistance): added for a user's reference.

c. Line 164: vector A is expanded to 15 000 words to permit double precision.

d. Lines 185, 188, 189, 234, 236, 242, 250: amended to conform to double precision.

e. Lines 245-249: information on array location in core is printed routinely.

2.2.2 AITKEN and Function Area

a. Line 9: TITLE is reduced to 18 words, or from 4 lines to 2 lines.

b. Function AREA is called by CHISQP.

2.2.3 ASSIGN

a. Line 12: REAL*4 variables are arguments for single precision IMSL subroutines.

b. Lines 19, 21-23: new COMMON blocks are for options.

c. Line 20: COMMON/C/ holds parameter names for EXTERNAL NORM, an external function needed for IMSL subroutine, GFIT.

d. Lines 28-30: required information for IMSL subroutines, GFIT and USHIST.

e. Line 31: time check is placed in operation.

f. Lines 77-79: switch MTIME set equal to 1 activates printing of "time to compute discriminant functions."

g. Lines 122-130: if ellipses are to be plotted, the probability of membership of each observation in each cluster is read into a storage file.

h. Lines 136-256: if cluster means and standard deviations of the original variates are desired, they are obtained by backtransformation of each datum.

i. Lines 262-279: preparation for IMSL subroutine calls are triggered when IHIS = 1. Twenty points are required for these equiprobable cells, and the number of cells is the number of obervations divided by 5, but not more than 12.

j. Lines 280-294: the arguments in the first IMSL subroutine have the following definitions:

(1) NORM: user-supplied external subroutine that provides the ordinate of the hypothesized cumulative distribution function.

(2) NUM: input number of equiprobable cells in which the observations are tallied.

(3) Y: input vector.

(4) II: length of input vector.

(5) CELLS: output vector of length NUM containing counts of observations that fall into the equiprobable intervals.

(6) COMP: output vector of length NUM containing components of chi-square statistic.

(7) CS: output chi-square statistic.

(8) IDF: number of hypothesized distribution function parameters estimated from the data, e.g., 2 (mean and standard deviation). On output, IDF is the number of degrees of freedom of CS.

(9) Q: output probability of chi-square statistic exceeding CS if the null hypothesis is true.

(10) IER: output error parameter.

If the 95 percent probability level is selected, $Q > 0.05$ indicates that the null hypothesis cannot be rejected. The null hypothesis is that the distribution of Y-values is not significantly different from normal.

k. Lines 295-296: the arguments in the second IMSL subroutine have the following definitions:

(1) CELLS: see j(5).

(2) NUM: see j(2).

(3) ISPACE: spacing between histogram base, i.e., 0, 1, or 4.

(4) WORK1: work vector of length NUM*(ISPACE + 1) + 4 or more.

(5) WORK2: work vector of length 1 + WORK1/4 or more.

(6) IWORK: work vector that can be set equal to 50.

(7) IER: see j(10).

Additional information is available in the IMSL manual.

2.2.4 CHISQP

This function is called by NORMIX subroutine and there are no changes.

2.2.5 DENSIT

There are minor changes in the dimension statement.

2.2.6 DUMPER

There are minor changes in the dimension statement.

2.2.7 EIGEN

a. Line 52: COMMON/D/ is inserted to pass the MERGE switch from the NAMELIST in INFORM.

b. Line 53: KP is added to the dimension statement to facilitate the printing of eigenvectors.

c. Lines 196-208: the IGEN switch can be set equal to 1 in INFORM to activate the printing of eigenvalues and eigenvectors during each of three iterations.

2.2.8 HISTOG

a. Line 13: COMMON/D/ is inserted to pass the MERGE switch from the NAMELIST in INFORM.

b. Lines 14-16: there are minor changes in the dimension statement.

c. Lines 95-96, 126, 130-131: the MERGE switch can be set equal to 1 in INFORM to activate the printing of the interesting parts of the merging (i.e., hierarchical grouping) procedure in plain language.

d. Lines 102-103: the remaining part of the merging procedure is relegated to "comment" in the program to avoid excessive printing.

2.2.9 INFORM

a. Lines 12-15: three COMMON statements are inserted to pass NAMELIST parameters to KMEAN (IHIER and IZQ), HISTOG (MERGE), EIGEN (IGEN), ASSIGN (IHIS, MTIME, IMAP, IPOL, SCAL, POWER, IWIND, NWIND), NORMIX (MTIME, IOVER, NTIME), INITLE (IMAP), OUTPUT (IMAP, IUNITS, IPROF, IBACK).

b. Line 16: COMMON/G/ is inserted to pass external preliminary estimates of statistics to REVALU.

c. Lines 17-19, 25-70: a NAMELIST is used to enter essential parameters and option switches; each name is defined in lines 25-66.

d. Line 23: IZQ is a counter for grouping iterations in KMEAN.

e. Lines 71-74: a mandatory table composing the number, name, abbreviated name (for output tables), and units for each variable is printed by calling TABLE. The IAA switch is set equal to 1 for column headings in the output consisting of abbreviated variable names and units. The IAB switch is set equal to 1 in TABLE for backtransformation of winds.

f. Lines 75, 117-118, 121: the format for the input data is read and printed.

2.2.10 INITLE

a. Line 10: COMMON/F/ is inserted to pass the IMAP switch from the NAMELIST in INFORM.

b. Lines 12-13: there are minor changes in the dimension statement.

c. Lines 27-30, 37-39: the IMAP switch can be set equal to 1 in INFORM to activate the storage of information for ellipse-plotting.

2.2.11 KMEAN

a. Line 31: COMMON/D/ is inserted to pass the IHIER and IZQ switches from the NAMELIST in INFORM.

b. Lines 32-33: there are minor changes in the dimension statement.

c. Line 246: IZQ is an iteration counter for grouping.

d. Line 266: on the third and last iteration, the IHIER switch can be set equal to 1 in INFORM to activate the printing of a map of the hierarchical grouping.

2.2.12 MAPPER

There are no changes in this subroutine, which is called by OUTPUT.

2.2.13 MOMENT

Lines 6-7: there are minor changes in the dimension statement.

2.2.14 NORMIX

a. Line 12: COMMON/E/ is inserted to pass the MTIME and IOVER switches from the NAMELIST in INFORM.

b. Lines 37-39: time check and printout activated by MTIME set equal to 1 are placed in operation.

c. Lines 76-78: time check is placed in operation.

d. Lines 79-82: the NTIME switch can be set equal to 1 in INFORM to activate the dumping (i.e., punching) of statistics when parameter LTIME is exceeded.

e. Lines 88-91: time check and printout activated by MTIME set equal to 1 are placed in operation.

f. Line 118: the IOVER switch can be set equal to 1 in INFORM to activate the overriding of the chi-square test for the null hypothesis that n + m clusters have the same likelihood as n clusters.

2.2.15 NROOT

There are no changes in this subroutine, which is called by ASSIGN.

2.2.16 OUTPUT

a. Line 10-11: COMMON/F/ is inserted to pass the IMAP, IUNITS, and IBACK switches from the NAMELIST in INFORM.

b. Lines 12-14: there are minor changes in the dimension statement.

c. Lines 17-18: the switch for equal covariances (NOR = 0) or unequal covariances (NOR = 1) is established.

d. Line 19: the unit number is set equal to 11 for the ellipse-plotting storage file.

e. Line 40: an optional call can be made to TABLE to print column headings in output tables.

f. Lines 46-53, 64-66: the IMAP switch can be set equal to 1 in INFORM to activate the storing of the number of clusters (NTP), NOR (2.1.16c), cluster variable means (MEAN), cluster variable standard deviations (S), and cluster correlation coefficients among variables (R) for ellipse-plotting.

g. Lines 68-71: the IBACK switch causes the SCAL terms to be printed if it is set > 0. The SCAL term for each variate is the amount to be deducted.

h. Line 73-74: the IPROF switch can be set equal to 1 in INFORM to activate the profile plot in PROFILE, which is outside the NORMIX package.

2.2.17 PRTTYP

a. Lines 9-10: normal exit from NORMIX is made through this subroutine, using a statement that is printed routinely.

b. Line 14: an end of file mark is placed on the storage file for ellipse-plotting.

c. Line 15: an end of file mark is placed on the storage file for profile-plotting.

d. Line 16: this line is not adaptable to the UNIVAC 1110, so execution does not pass this point.

2.2.18 RELIST

There are no changes in this subroutine, which is called by KMEAN.

2.2.19 REVALU

a. Line 13: COMMON/G/ is inserted to pass external preliminary estimates of statistics from INFORM.

b. Lines 14-15: there are minor changes in the dimension statement.

c. Line 44: the four parameters related to preliminary estimates are printed.

2.2.20 SYMINV

There are no changes in this subroutine.

2.2.21 TYTLE

a. Lines 7, 19: the format is reduced to a 2-line printing of the title.

2.2.22 UPLOW

There are no changes in this subroutine, which is called by HISTOG.

2.2.23 PUTMSG

There are no changes in this subroutine, which is called by MAIN.

2.2.24 NORM

This new subroutine, required by the IMSL subroutine, GFIT, is used in ASSIGN.

2.2.25 TABLE

This new subroutine prints a table of variable numbers, names, abbreviated names, and units. It also prints optional column headings in the output. It is called by INFORM, OUTPUT, and ASSIGN, using the switch, IUNITS. The four items supplied for a table should be right-justified in columns 10, 30, 40, and 50.

2.2.26 PROFILE

This new subroutine stores data for an optional plot of profiles of statistics for each variable and up to three clusters. It is called by OUTPUT, using the switch, IPROF.

2.2.27 SEED

This new subroutine obtains the clock time in thousandths of a second to use as an initial random number seed value. The value is printed so that the subsequent sequence of pseudo-random numbers can be repeated, if desired, by entering the same initial seed value.

2.2.28 SEED1

This assembly language subroutine supplies the clock time to SEED for use as a seed for pseudo-random numbers.

3.0 OPTIONS IN NORMIX PROGRAM AND NORMIX FLOW

A substantial number of options are available to a user of the NORMIX program. Some of the options are in the form of essential selections among various possibilities; these options are continued from earlier versions of the program. Other options consist of features that can be turned on by a switch; these options have been created in 1979 during the cluster analysis effort made by the individuals named in Section 1.

The general flow of the NORMIX program, including its options, is shown in Figure 1. This diagram depicts the three parts: NORMIX-PREP, NORMIX, and CALCOMP. The first part consists of new options that are incorporated in a separate preparatory subprogram called NORMIX-PREP. The second part consists of grouping and clustering with all of the old options and a few new ones; these features compose the NORMIX program that was discussed in Section 2. The third part contains CALCOMP and Tektronix plotting routines that are activated to display information related to data and statistics that are stored optionally in NORMIX. The various options are now treated individually in the order observed in Figure 1.

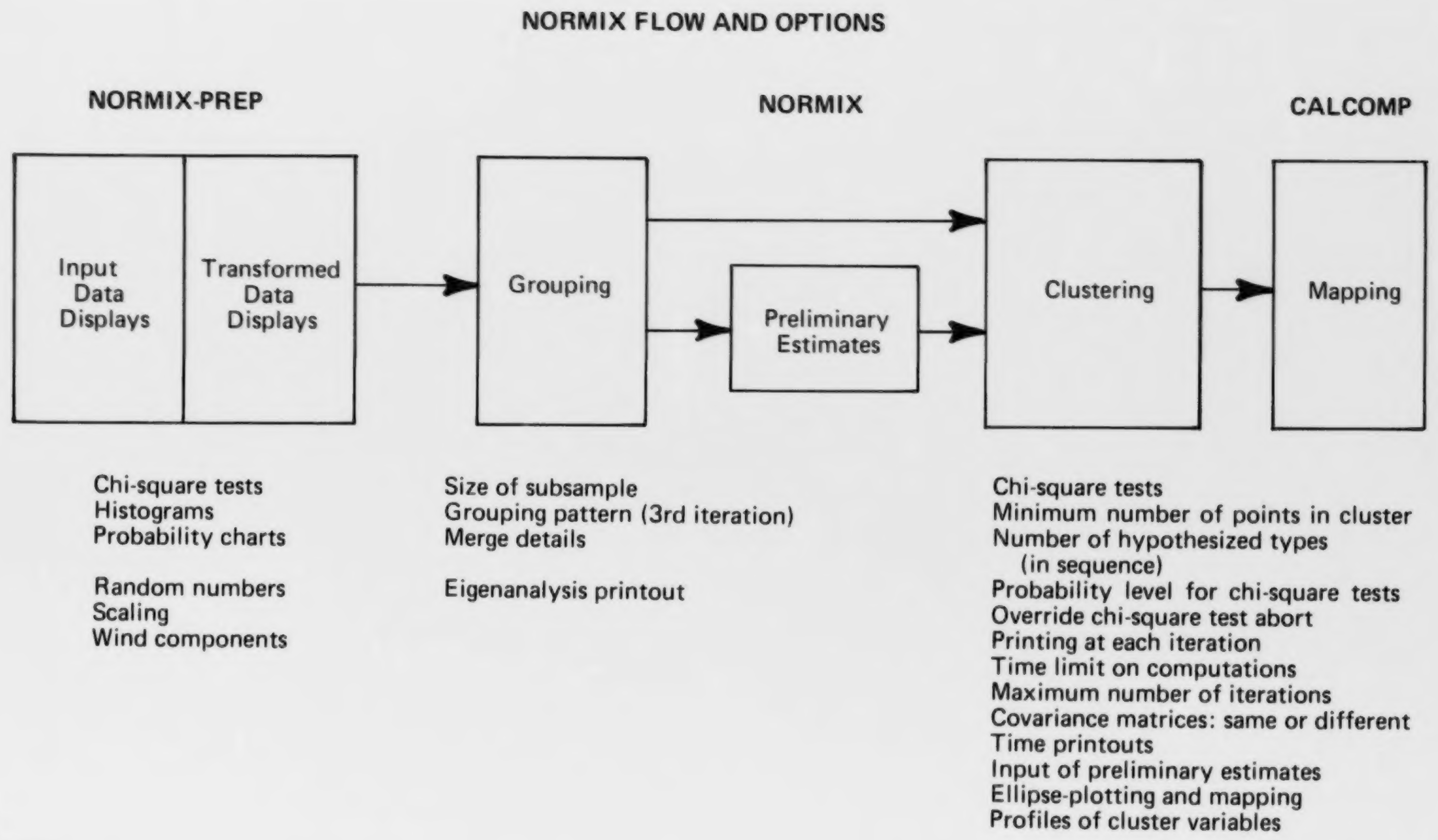

NORMIX FLOW AND OPTIONS
NORMIX-PREP
NORMIX
CALCOMP
Input Data Displays
Transformed Data Displays
Grouping
Preliminary Estimates
Clustering
Mapping
Chi-square tests
Histograms
Probability charts
Random numbers
Scaling
Wind components
Size of subsample
Grouping pattern (3rd iteration)
Merge details
Eigenanalysis printout
Chi-square tests
Minimum number of points in cluster
Number of hypothesized types (in sequence)
Probability level for chi-square tests
Override chi-square test abort
Printing at each iteration
Time limit on computations
Maximum number of iterations
Covariance matrices: same or different
Time printouts
Input of preliminary estimates
Ellipse-plotting and mapping
Profiles of cluster variables

Figure 1. General flow of NORMIX program and its associated routines with options listed underneath.

3.1 NORMIX-PREP OPTIONS

Two passes are made through NORMIX-PREP when this subprogram is accessed to perform one or more of the available options. During the first pass, any of the options under NORMIX-PREP in Figure 1 can be activated. During the second pass, which deals with the transformed data from the first pass, only the first three options dealing with chi-square tests, histograms, and probability charts can be exercised. A listing of the NORMIX-PREP program is included in Appendix B.

3.1.1 Chi-Square Tests

Tests are made of the null hypothesis that the input data, or the transformed data, are distributed normally, as tested by the IMSL subroutine, GFIT. If the output probability of the chi-square statistic exceeding chi-square is less than 0.05, the null hypothesis can be rejected. This "option" cannot be switched off by the input parameters.

3.1.2 Histograms (IHIST)

Equal probability histograms for the input data and the transformed data are displayed, using the IMSL subroutine, USHIST. These histograms are intimately related to the chi-square test in that equally long bars across the chart indicate normality. Conversely, a piling up of frequencies in the center denotes a departure from normality in the form of excessive kurtosis. Furthermore, various kinds of skewness and multimodality can be shown. This option is turned on by IHIST = 1.

3.1.3 Probability Charts (IPROB)

Another test for normality of a distribution is a plot of the data on normal probability graph paper with a visual check of the straightness of the resulting line connecting adjacent points. This test is performed by computer plotting, using an internal subroutine called CUMPLT. The plot is activated by IPROB = 1.

3.1.4 Pseudo-Random Numbers (IRAND, FACTOR, IQOL)

Collinearity is a nemesis in matrix inversion or the solution of simultaneous equations. If collinearity is suspected, the user should take some action to take care of the problem. In a crude way collinearity can often be alleviated by adding a pseudo-random number to each datum. The IMSL subroutine, GGNQF, is selected for this purpose. GGNQF provides a normal (0,1) deviate that is multiplied by two factors: the first factor is merely the standard deviation of the variate and the second factor is an empirical scaling factor, FACTOR, that is set by the user in the NAMELIST. Experimentatation with various factors has revealed that 10^{-5} is satisfactory for pollutant data. This value is the default value for FACTOR. When IRAND is set equal to 1, a pseudo-random normal deviate is added to all variables. When IRAND is set equal to 2, a pseudo-random normal deviate is added to the variates indicated by the list IQOL.

An initial seed value is obtained by Subroutine SEED, which takes the current clock value for this purpose from an assembly language subroutine, SEED1.

3.1.5 Scaling (THRESH, MINPOL)

When a variate is known or suspected to be distributed logarithmic normally, the user may elect to transform by taking logarithms. If the data have zeroes or negative values embedded in the set, scaling is necessary. Scaling is accomplished by adding a constant to each observation for the variate being scaled. More specifically, in NORMIX-PREP a threshold value (THRESH) is set for each variable, as well as a minimum value (MINPOL) so that when the threshold value is reached or surpassed on its lower side, the minimum value is changed to MINPOL and all other values for the variates are increased equally. Scaling is done in a similar way for power transformations.

3.1.6 Wind Components (NWIND, IWIND)

Wind direction is routinely measured from true north and recorded with wind speed. If the components of the wind from the west (positive) and south (positive) are desired, the switch is turned on by NWIND = 1. Then the variable for wind speed is designated by IWIND(1) and the wind direction variable is designated by IWIND(2). These variables are transformed by NORMIX-PREP into the west component and south component, respectively. If another pair of orthogonal components is sought, the axes are rotated clockwise a number of degrees, as indicated in the NAMELIST by DEG. Then the components obtained are positive-valued at angles of 180° and 270° from the new zero for the wind direction.

3.2 NORMIX OPTIONS FOR GROUPING PHASE

There are four options under the "Grouping" box in Figure 1. These options are now explained in some detail.

3.2.1 Size of Subsample (MAXAV)

The number of observations to be considered in hierarchical grouping is set by MAXAV. This quantity defaults to 150 or to the sample size N if $N < 150$. If MAXAV is set to a value smaller than N, then changes in the order of data input may produce distinct results in the output. These results are caused by the "local maxima" of the likelihood function that are reached during the convergence process.

3.2.2 Grouping Pattern (IHIER)

If the grouping pattern consisting of dots and X's is to be displayed, the switch is turned on by IHIER = 1. This pattern reveals the order of merging of groups. Only the third and final iteration can be printed; this map becomes an input for the main clustering algorithm in NORMIX. A detailed explanation of the mapping symbolism is provided by Crutcher (1980).

3.2.3 Merge Details (MERGE)

The grouping pattern that can be printed optionally is described more explicitly by merge statements that are switched on by MERGE = 1. Even so, only the initial and final series of statements are printed, the remainder are considered to be of negligible interest.

3.2.4 Eigenanalysis Printout (IGEN)

The eigenvalues and eigenvectors of the covariance matrix for the entire data set can be printed for all three iterations. The switch is turned on by IGEN = 1.

3.3 NORMIX OPTIONS FOR CLUSTERING PHASE

There are numerous options under the "Clustering" box in Figure 1. These options are now discussed.

3.3.1 Chi-Square Tests and Histograms (IHIS)

Analogous to Section 3.1.1, tests can be made of the null hypothesis that the cluster variates are not significantly different from a normal distribution. This option is activated by IHIS = 1. When equal probability histograms are coupled with this option, they can be displayed for cluster variates.

3.3.2 Minimum Number of Points in Cluster (ENMIN)

The default value for this parameter is the number of variates plus one. If the number of observations is very small, e.g., less than 20, use of the default value may cause a premature cutoff in the output. To avoid this difficulty, ENMIN should then be set at a lower value, such as 2.

3.3.3 Number of Hypothesized Types (NTYP)

When a given number of clusters, or types, have been identified, a chi-square test is conducted to test the null hypothesis that the present clustering is not significantly different from the previous clustering. NTYP is a sequence of integers representing successive trials. The integers must increase monotonically, but gaps are permitted in the sequence.

3.3.4 Probability Level for Chi-Square Test (HTEST)

Selection of a probability level, e.g., .05 or .01, establishes the decision level for rejection of the null hypothesis. If the "probability of null hypothesis" appearing in the printout is less than HTEST, the program rejects the null hypothesis and proceeds to the next integer in the NTYP sequence. If rejection is not accomplished, inspection of the output reveals how close it came to rejecting the null hypothesis.

3.3.5 Override Chi-Square Test Abort (IOVER)

If termination by failure to reject the null hypothesis is not satisfactory to the user, the abort can be avoided by a switch called IOVER. The switch is turned on by IOVER = 1.

3.3.6 Printing at Each Iteration (IPRINT)

If a printout of proportions and statistics for each cluster at each iteration in the iterative solution of the maximum likelihood equations is wanted, this information is secured by setting IPRINT = 1.

3.3.7 Time Limit on Computations (LTIME)

To avoid a termination by expiration of run time while computation or printing is in progress, LTIME can be set equal to run time -1 in minutes. This process permits the machine to punch a set of cards for input as preliminary estimates and to continue the output sequence in another run. However, switch NTIME must be activated (by NTIME = 1) to permit cardpunching. In addition, a number of cards must be made available, as would be indicated on the run card.

3.3.8 Maximum Number of Iterations (ITERX)

The iterative solution of the maximum likelihood equations may converge slowly, requiring many iterations.

3.3.9 Correlation Matrices: Same or Different (NOR)

If the clusters corresponding to an NTYP are constrained to have identical correlation matrices, the maximum likelihood estimates of the parameters are distinct from the unequal covariance case. The distinctions become apparent when ellipses are drawn at a given probability level for the clusters. In the case of "same variances," the ellipses have similar sizes and orientations. A mapping of data points based upon discriminant analysis is displayed for all clusterings as great as 3, but only in the case of common covariances. Switch NOR = 0 provides common correlation matrices, and NOR = 1 gives different covariance matrices.

3.3.10 Time Printouts (MTIME)

The time expended in certain operations can be printed by setting a switch called MTIME equal to 1. The operations are: hierarchical grouping, iteration of maximum likelihood equations, and discriminant functions.

3.3.11 Backtransformation Statistics (IBACK, IPOL, NWIND, IWIND)

If a table of backtransformed statistics is desired, switch IBACK is set equal to 1. Pollutant variate numbers are established by IPOL, and winds are changed from components to speed and direction by setting NWIND equal to 1 and

IWIND equal to the variate numbers. Each observation is backtransformed in the process. Additional information is available in Appendix F, Section 2.

3.3.12 Input of Preliminary Estimates

If the preliminary estimates from hierarchical grouping are to be overridden by estimates of means, standard deviations, and correlations for the variables in each cluster, the imposed values are read immediately following the reading of the data input in the runstream. An initial card or file line consisting of four integers must always be present. If no preliminary estimates are to be read, the first integer must be zero, i.e., the line can be blank. Otherwise, the initial card reads in (1) a nonzero integer (IVALUE) to indicate that such estimates are forthcoming and to give the number of clusters involved, (2) a switch (IMN) that is set equal to zero to indicate that no mean values are to be read, (3) a switch (ISTD) that is set equal to zero to indicate that no standard deviations are to be read, and (4) a switch (ICOR) that is set equal to zero to indicate that no correlation coefficients are to be read. IVALUE is set nonzero in INFORM, making the 4-integer or blank card mandatory. The format for the four integers is (4I5). If any of the integers are nonzero, corresponding values are read with the following formats:

1A	proportion of population in cluster 1	(F8.8)
1B	means of variables in cluster 1	(10F8.4)
1C	standard deviations of variables in cluster 1	(10F8.4)
1D-1	first row of correlations in cluster 1	(10F8.4)

1D-2	second row of correlations in cluster 1	(10F8.4)
⋮		⋮
1D-n	nth row of correlations in cluster 1	(10F8.4)
2A	proportion of population in cluster 2,	(F8.8)
	etc., for remaining clusters.	

If common covariances are used, then all of the proportions and means are read first, and a single set of within-type standard deviations and correlations are read second. This ordering and format are followed for either correlation/covariance option in the punching of cards initiated in Subroutine DUMPER by the switch, NTIME, and the parameter, LTIME.

3.4 CALCOMP OPTIONS

There are two kinds of displays available under the CALCOMP heading and mapping box of Figure 1. These displays and their driver programs are discussed in Sections 4.3 and 4.4.

4.0 DISCUSSION OF NORMIX PROGRAM AND SUBPROGRAMS

As has been noted in Section 3, three subprograms are associated with the NORMIX program. The preparatory subprogram called NORMIX-PREP will be discussed in Section 4.1; the NORMIX program will then be detailed in Section 4.2; the subprogram that produces cluster variate profiles, called PROFILE, will be discussed in Section 4.3; and the ellipse-plotting routine called ELLIPSE will be described in Section 4.4.

4.1 NORMIX-PREP

The optional data preparation phase of NORMIX consists of a file called NORMIX-PREP with five subroutines. Since all of the program routines are devised to implement an option, additional discussion would repeat the material in Section 3.1. A brief statement of the purpose of each subprogram will avoid such repetition.

4.1.1 MAIN

There are two passes through MAIN. In the first pass, after reading and printing the data (lines 36-52), the mean and standard deviation of the sample for each variate are computed (lines 68-78) for the chi-square test. Then random numbers can be applied to any or all of the variables (lines 87-100), and IMSL subroutines can be invoked for the chi-square test and for histograms of the original data, as modified by random numbers, if any (lines 119-138). If a probability chart is desired for each variate, the data are placed in a monotonic, increasing sequence and Subroutine CUMPLT is called (lines 143-162). Then the data values for each variate are scaled as needed for a transformation by natural logarithms or powers (lines 168-200), and information is stored (line 244) for possible backtransformation in the NORMIX program. If wind speed and direction are to be converted to west and south components (lines 213-218), the components can undergo a rotation to another set of orthogonal axes (lines 223-226).

In the second pass, the data are printed again with transformed variables replacing original values and they are stored by unit number 10. Then the IMSL subroutines are called again, if desired, for a chi-square test and histograms.

4.1.2 CUMPLT

As a supplementary test of the normality of a distribution, a cumulative distribution chart can be produced with data cell values plotted on a linear ordinate axis versus cumulative cell frequencies on the abscissa on a normal probability scale. The straightness of a line connecting the plotted points is a measure of normality. The arguments for the subroutine call are identified in MAIN, lines 158-160.

4.1.3 NORM

Subroutine NORM is an external function needed for the IMSL chi-square subroutine, GFIT. Its two arguments are the mean and standard deviation of the variate sample to be tested.

4.1.4 SEED

Subroutine SEED is called to obtain a seed value for the IMSL subroutine GGNQF, which generates pseudo-random numbers. The seed value is itself randomized by a call from SEED to SEED1, an assembly language subroutine that picks up the current clock value in hundredths of a second. This value is printed to assure uniqueness among runs and to permit exact duplication of pseudo-random numbers, if desired. However, duplication of a set of these numbers can only be arranged by programming to override the new seed value obtained in MAIN, line 93.

4.2 NORMIX

The NORMIX flow chart presented in Figure 2 includes the satellite routines referred to in preceding sections as NORMIX-PREP, ELLIPSE, and PROFILE. However, its main purpose is to give a more detailed picture of the flow in NORMIX itself. The flow will be discussed by the nine divisions to be found in Figure 2, moving clockwise from a starting position near 2:00 o'clock.

4.2.1 NORMIX-PREP

As pointed out in Section 4.1, NORMIX-PREP is an optional data preparation package. Its potential usefulness for examining, testing, and transforming a data set is presented therein.

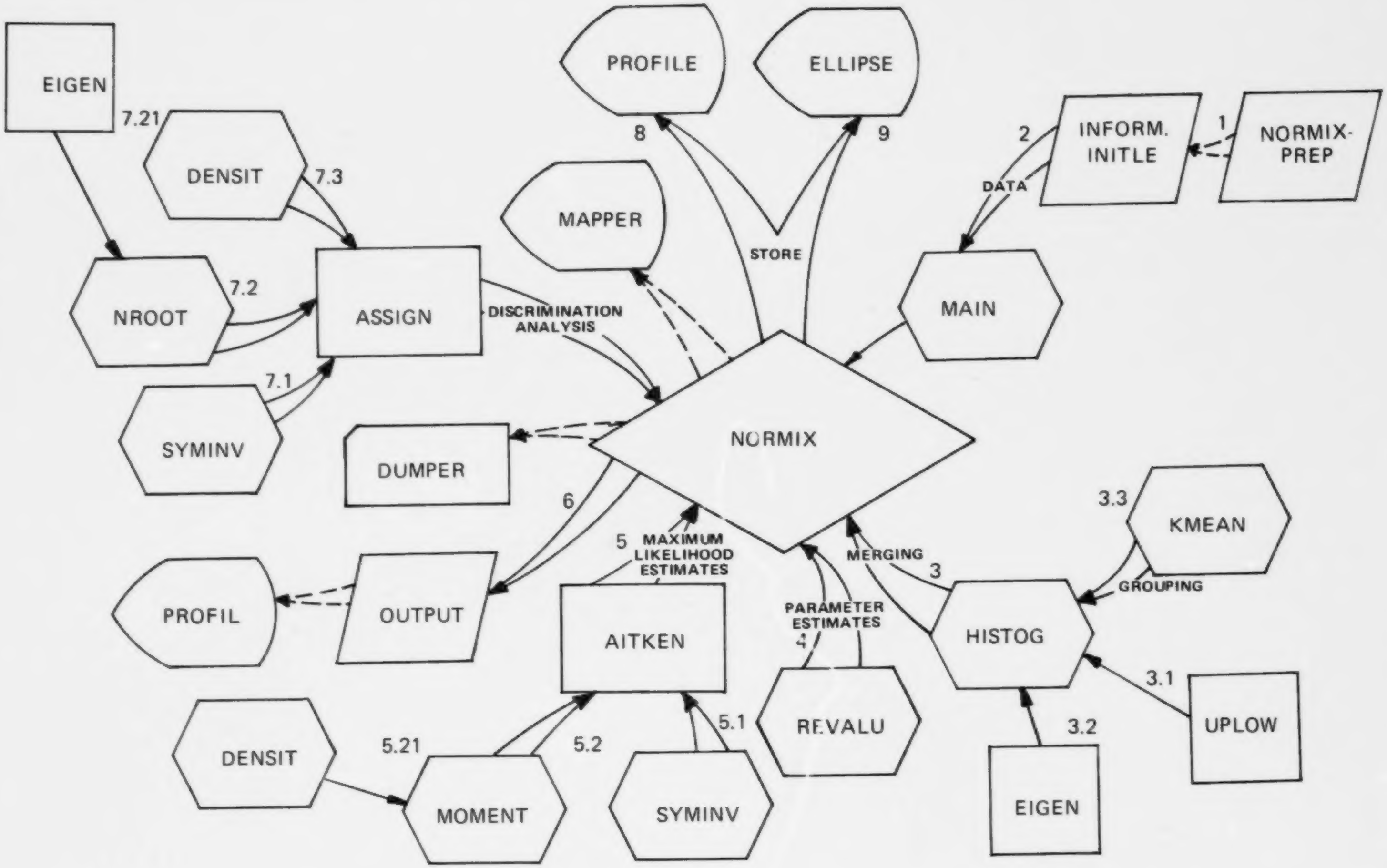

EIGEN
7.21
DENSIT
7.3
NROOT
7.2
ASSIGN
7.1
SYMINV
DISCRIMINATION ANALYSIS
PROFILE
8
ELLIPSE
9
MAPPER
STORE
INFORM. INITLE
2
DATA
1
NORMIX-PREP
MAIN
NORMIX
DUMPER
6
5
MAXIMUM LIKELIHOOD ESTIMATES
PROFIL
OUTPUT
AITKEN
PARAMETER ESTIMATES
4
MERGING
3
3.3
KMEAN
GROUPING
HISTOG
3.1
UPLOW
3.2
EIGEN
REVALU
5.1
SYMINV
DENSIT
5.21
MOMENT
5.2

Figure 2. Detailed flow of NORMIX subroutines and associated routines.

4.2.2 INFORM, INITLE

These two NORMIX subroutines are essentially data-entering subprograms. INFORM uses the NAMELIST convention to provide the parameters for the numerous NORMIX options; it initializes various quantities and sets up 5 COMMON blocks for transmission of information; it calls Subroutine TABLE to read and to print information about the names and units of the variates; and finally, it prints some of the parameters in their final form ready for processing.

Following the execution of INFORM, control is passed to MAIN, which allocates storage dynamically to most parameters. Then subroutine NORMIX, the hub through which the computations flow, is accessed. NORMIX first calls INITLE, which reads the input data array and computes the arthmetic mean value of each variate. This subroutine also computes the covariance matrix for the input data.

4.2.3 HISTOG

Preliminary estimates of data clustering are obtained in Subroutine HISTOG by calling Subroutines UPLOW (to rearrange the covariance matrix in lower triangular form), EIGEN (to compute the eigenvalues and eigenvectors of the same matrix), and KMEAN. The latter subroutine combines "MacQueen's KMEANS clustering procedure" (1967) with "Ward's hierarchical grouping procedure" (1967). The MacQueen method operates on an assigned number of data values from the array, i.e., 150 or fewer.

Each value initially constitutes a "cluster" and intercluster distances are computed so as to combine the nearest pair. A new datum is entered into the space left by the merger and all distances are again computed so as to join the nearest pair. This process continues until all the data have been read. Then Ward's method ensues, minimizing the variance of the sum of Mahalanobis distances between points within a cluster. Wolfe (1971) modifies Ward's procedure by the use of three iterations to adjust for the within cluster variance:

a. Transform data to principal axis factor scores by Equation 1.

$$Y = L^{-1/2} F^{T} X \tag{1}$$

where Y = principal axis factor score matrix

L = diagonal matrix of eigenvalues

F^T = transpose pf eigenvector matrix obtained from the covariance matrix

X = input data array

b. Compute the variance to be minimized by Equation 2.

$$d_{ij}^{2} = (Y_i - Y_j)^{T} (Y_i - Y_j) \tag{2}$$

where the i and j are points in the same cluster.

c. Determine the hierarchical grouping for minimum variance. This step completes Iteration 1.

d. Find the within-group covariance matrix for 10 clusters.

e. Compute a variance to be minimized by Equation 3.

$$d_{ij}^2 = (X_i - X_j)^T C_2^{-1} (X_i - X_j) \tag{3}$$

where C_2^{-1} = inverse of within-group covariance matrix for 10 clusters.

f. Determine a new hierarchical grouping for minimum variance. This step completes Iteration 2.

g. Find a new within-group covariance matrix for 10 clusters, C_3.

h. Compute a final variance to be minimized by Equation 3.

i. Determine a third and final hierarchical grouping for minimum variance. This step completes Iteration 3.

If the assigned number of data points in the MacQueen clustering procedure is equal to the number of points in the input data array, N, then the entire NORMIX output should remain unchanged, regardless of the input data sequence. However, if the assigned number of points is less than N, a reversal, randomization, or other alteration in the input data sequence may cause differences to appear in the outputs. Experience to date has not disclosed any large differences. The map of the hierarchical grouping is a printer display described in detail in Volume I.

4.2.4 REVALU

The hierarchical grouping of Section 4.2.3 is executed once per run, but REVALU Subroutine provides an entry point for independent preliminary estimates of any combination of the following: mixing proportions, means, standard deviations, and correlation coefficients for the variables in each cluster. Since REVALU is called each time the number of hypothesized types is increased, independent preliminary estimates can be entered repeatedly. REVALU always prepares the initial estimates for iteration by computing cluster proportions, means, and covariances from the results of hierarchical grouping. If independent estimates are entered, they override these computed values.

4.2.5 AITKEN

The maximum likelihood equations (Wolfe, 1971) to be solved for estimates of the proportions, means, and covariances of each cluster are Equations 4 to 6, given below:

$$\hat{\lambda}_s = 1/n \sum_{k=1}^{n} \hat{P}(s|X_k) \qquad (4)$$

where $\hat{\lambda}_s$ = estimate of mixing proportion for type s

n = number of points in type s

$\hat{P} = (s|X_k)$ = probability of membership of X_k in type s;

$$\hat{u}_s = 1/n\hat{\lambda}_s \sum_{k=1}^{n} X_k \hat{P}(s|X_k) \qquad (5)$$

where $\hat{u}_k$ = vector of means for variates in type s

X_k = vector of variate values for kth sample point;

$$\hat{C}_s = 1/n\hat{\lambda}_s \sum_{k=1}^{n} (X_k - \hat{u}_s)(X_k - \hat{u}_s)^T \hat{P}(s|X_k) \qquad (6a)$$

where $\hat{C}_s$ = covariance matrix for type s.

If the assumption of identical cluster covariances is invoked, Equation 6 should read:

$$\hat{C} = 1/n\hat{\lambda}_s \sum_{k=1}^{n} x_k x_k^T - \hat{u}_s \hat{u}_s^T . \qquad (6b)$$

Because there are unknowns on both sides of these equations, an iterative solution is adopted. The Newton-Raphson iteration has yielded to the Aitken method in the NORMIX program, hence Subroutine Aitken is called from NORMIX during each iteration in loop 5 (lines 51-84). The programmed criterion for convergence is set at 10^{-5}. If this criterion is met, iteration ceases.

There are three ways for execution to cease at this point. The first way is the "no convergence" case when the likelihood fails to exceed the maximum likelihood of prior iterations in nine consecutive attempts. The second way is to encounter a zero pivot in one of the covariance matrix inversions that is accomplished by calling SYMINV. The third way is to reach the time limit for processing, if this option is on, so as to dump the parameter statistics on punched cards. In addition to these abortive cases, a warning is printed if one of the clusters has fewer than the assigned minimum number of observations.

4.2.6 OUTPUT

To tabulate the means, standard deviations, and correlation coefficients for each cluster at the beginning and end of the iteration process, Subroutine OUTPUT is called from NORMIX immediately after final exit from REVALU and AITKEN, respectively. Exit from REVALU is the "iteration 0" case. However, for the equal covariance case, just the means are printed, plus within-group standard deviations for each variate and the within-group correlation coefficients. Statistics for the entire data set are printed each time routinely, as they are invaluable to the investigator in the examination of his results. Column headings showing abbreviated variate names and units can be inserted by an optional call of Subroutine TABLE in OUTPUT. Subroutine PROFILE can also be called so as to store information for profile plots of cluster variates. This option will be explored in Section 4.3.

4.2.7 ASSIGN

The final operation of the NORMIX program is a discriminant analysis and listing of cluster membership, accompanied by cluster statistics in terms of the transformed data and (optionally) in terms of the original data. This operation is accomplished by Subroutine ASSIGN and its satellite subprograms, SYMINV, NROOT, and DENSIT, with an assist from EIGEN. The listing is convenient, as it reveals which cluster claims a particular observation and whether or not the probability of membership is divided among clusters, and to what extent quantitatively. Total agreement is not necessary between cluster membership count and final iteration count, but experience to date indicates that it is usually attained.

Discriminant function centroids, or "discriminant scores," as explained in Cooley and Lohnes (1971), are printed for the common covariances case, which is linear. The unequal covariances case is quadratic in this respect, so such information is not available. Likewise, the mapping of all possible pairs of discriminant functions by computer printing is only possible in the common covariances case. This subroutine is denoted by MAPPER in Figure 2.

IMSL subroutines are called by ASSIGN to perform chi-square tests for normality and draw histograms of cluster distributions.

4.3 CLUSTER VARIATE PROFILES

As was indicated in Section 4.2.6, the user may store information for an external plot of the profiles of cluster variates 2 or 3. When IPROF has been set equal to 1 in the INFORM NAMELIST, Subroutine PROFILE stores cluster proportions, ±3 standard deviation limits around the overall mean for each variate, and also the mean and ±2 standard deviations for each cluster variate. The latter trio of values are in terms of relative (background) distance on the transformed data scale that pertains to the overall mean and 3 standard deviations. If a logarithmic transformation has been used, then the mean values are now geometric means and the standard deviations are now geometric standard deviations. The geometric means are equivalent to median values for the original data set and cluster subsets.

As shown in Figure 2, the profiles are displayed via CALCOMP, Tektronix, or other plotting device, using a driver program called PROFILE. An example of the product is given in Figure 3, which is an adaptation of a tricolor CALCOMP plot. In this chart, the three horizontal lines across the page represent the medians of the transformed data (i.e., the geometric means of the data after logarithmic transformation) for the eight variates and the 3-sigma limits equidistant from the geometric means, with the means lying on the middle line. The three cluster proportions are printed at the left side of the curves, the geometric mean cluster value for each variate is plotted on the vertical scale already established for each variate, and the 2-sigma

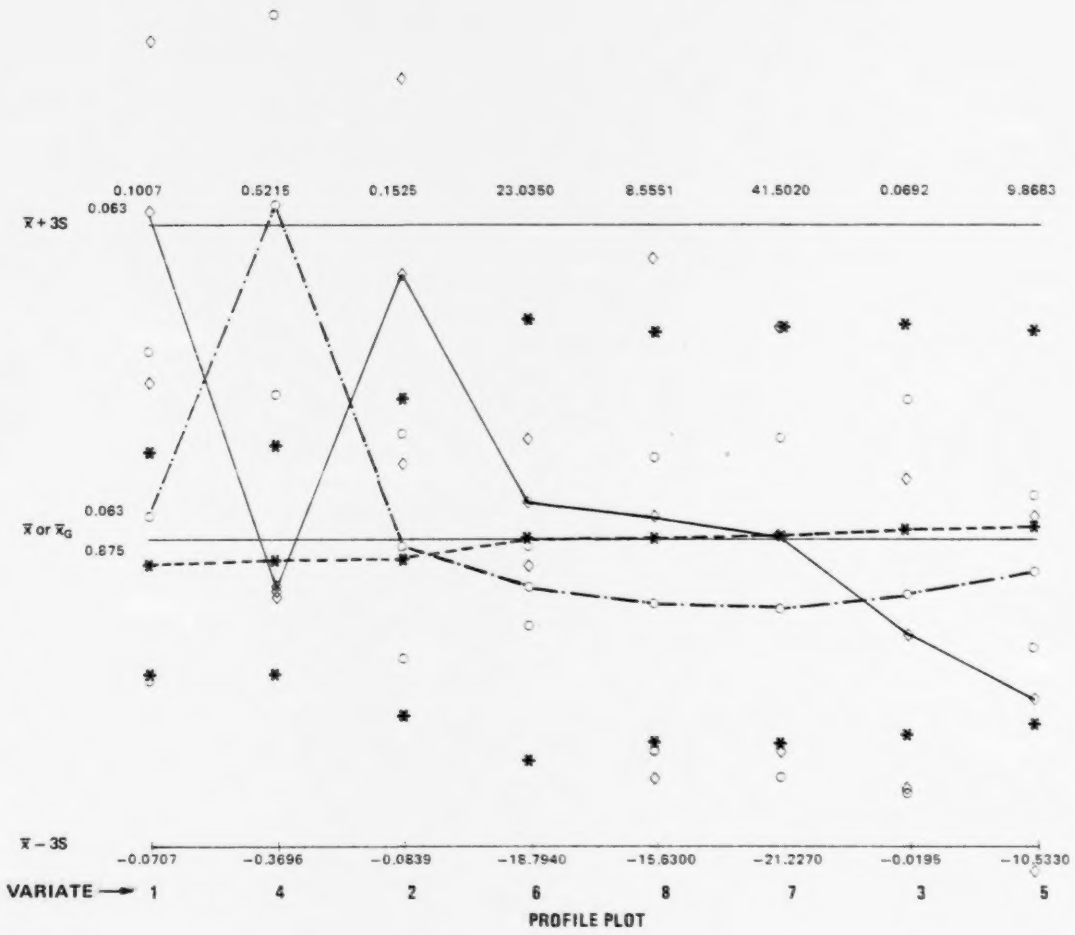


Figure 3. Cluster variate profile plot for the three clusters with eight reordered variates.

limits around each of the cluster means are also plotted. Finally, the 3-sigma limits are affixed for each variate in terms of the original data set. The deductible constant for the backtransformation is called SCAL and is subtracted from each transformed variable after the antilogarithm or reciprocal power has been taken. The variables can be rearranged so that the largest cluster has a rising curve on the chart. This option is turned on by setting IFLAG = 0 in MAIN. A listing of PROFILE is presented in Appendix C.

4.4 PROBABILITY ELLIPSES

An ellipse can be drawn for the unequal covariances case to represent the bivariate distribution of projections of n-dimensional data on a selected pair of axes. Such an ellipse has a given probability, so that it encases a given fraction of points in a cluster when the cluster has a normal distribution. Its location is determined by the centroid, the orientation of the major axis by the variances and covariances of the two variates, and the lengths of the major and minor axes depend upon the eigenvalues of the covariance matrix. Examples of ellipse-plotting for various sets of meteorological data have been presented by Crutcher and Joiner (1977).

When IMAP has been set equal to 1 in the INFORM NAMELIST, information for ellipse-drawing is stored by Subroutines INITLE, OUTPUT, and ASSIGN. Specifically, INITLE stores the number of variables, the number of observations, and the entire transformed data set for mapping data points on the ellipses; OUTPUT stores the means, standard deviations, and correlation coefficients for the entire data set and for each cluster as sets of clusters are tested in sequence; ASSIGN stores the cluster membership indicator for each data point. The user should note that NORMIX-PREP stores the scaling terms at the end of the transformed data set, and this additional line (or lines) would be superfluous to ellipse-plotting. Since wind scales are reversed to conform to convention, a pollutant plotted against a wind component must also be reversed. A listing of ELLIPSE appears in Appendix D.

4.5 BACKTRANSFORMATION STATISTICS

Any transformation of the input data, such as by taking natural logarithms, square roots, etc., leads to overall statistics and cluster statistics in NORMIX in terms of transformed data. The original data are lost in the first transformation in NORMIX-PREP. If the user wants to see the means and standard deviations in terms of the original data, he can turn on a switch in INFORM called IBACK, setting IBACK = 1 to perform the computations. When this switch is on, the scaling constant, SCAL, for each transformed value in NORMIX-PREP, is read by INITLE from the end of the file of transformed data, using a 6E12.5,/,6E12.5-format. After the antilogarithm or reciprocal power of the original data has been taken, the scaling constant is subtracted from each data value in a cluster. The statistics for the backtransformed data are computed and printed in ASSIGN. Subroutine TABLE can be utilized to obtain appropriate column headings for backtransformed data, including wind speed and direction.

4.6 PARAMETER LISTINGS AND OTHER USER AIDS

The NORMIX program with its recent modifications and added options is available at EPA's National Computing Center at Research Triangle Park, North Carolina. To assist the user, lists of parameters, their default values, and other pertinent information are placed in Appendix E for the NORMIX-PREP, NORMIX, PROFILE, and ELLIPSE programs. Illustrative outputs follow in Appendix F. Further information can be obtained by contacting the author at (919) 549-0611.

4.7 RECOMMENDATIONS FOR FUTURE WORK

The recent advances made in the field of cluster analysis as reported by Crutcher (1980) and in this report can be enhanced by performing the following steps, not necessarily in the order given:

a. Incorporate IMSL subroutines within NORMIX and NORMIX-PREP subroutines to permit greater flexibility to the user who does not have access to the IMSL package.

b. Continue the investigation of time series of ellipses over hours of the day, days of the week, and seasons. The product of this investigation is the mapping of cluster members when ozone is plotted against a wind component or when wind components are plotted against each other. This operation is currently being done for CHAMP data from Angwin, California; Loma Linda, California; and Magna, Utah.

c. Resume the investigation of the MIKCA (Multivariate Iterative K-means Cluster Analysis) algorithm as a source of preliminary estimates for NORMIX.

d. Investigate other cluster analysis programs, such as those belonging to IMSL, SAS, SPSS, and BMDP, as well as CLUSTAN. Compare the results of these programs with those obtained from NORMIX.

e. Incorporate a suitable version of the McGill "Super-Duper" Random Number Package with NORMIX-PREP, attaining as much interoffice flexibility as possible.

f. Experiment with 3-dimensional CALCOMP plots as improved displays of the mapping of cluster variates.

g. Examine the assumptions inherent in the NORMIX and NORMIX-PREP programs regarding multivariate normality, observing the outcome of systematic violation of assumptions.

h. Investigate the oddities that occur in the classification of historical Fisher Iris data by NORMIX and other clustering programs.

5.0 REFERENCES

Cooley, W. W., and P. R. Lohnes. 1971. Multivariate Data Analysis. John Wiley and Sons, New York, New York. 364 pp.

Crutcher, H. L., and R. L. Joiner. 1977. Separation of Mixed Data Sets into Homogenous Sets. NOAA Technical Report EDS 19, National Climatic Center, Asheville, North Carolina. 167 pp.

MacQueen, J. 1967. Some Methods for Classification and Analysis of Multivariate Observations. Proceedings of the Fifth Berkeley Symposium of Mathematical Statistics and Probability, Vol. 1. University of California Press, Berkeley, California. pp. 281-297.

Rhodes, R.C., H.L. Crutcher, B. B. Fairbairn, C. Nelson, and M. Symons. 1980. Applications of Cluster Analysis to Aerometric Data. U.S. Environmental Protection Agency, Research Triangle Park, North Carolina (To be published in the EPA Research Reporting Series). 136 pp.

Ward, J. H., Jr., K. Hall, and J. Buckhorn. 1967. PERSUB Reference Manual. PRL-TR-07-3-(11), Personnel Research Laboratory, Lackland Air Force Base, Texas.

Wolfe, J. H. 1965. A Computer Program for the Maximum Likelihood Analysis of Types. U. S. Naval Personnel Research Activity, Technical Bulletin 65-15, San Diego, California. 52 pp.

Wolfe, J. H. 1967. NORMIX: Computational Methods for Estimating the Parameters of Multivariate Normal Mixtures of Distributions. U. S. Naval Personnel Research Activity, Research Memorandum SRM 69-2, San Diego, California. 30 pp.

Wolfe, J. H. 1971. NORMIX 360 Computer Program. Naval Personnel and Training Research Laboratory, Research Memorandum SRM 72-4, San Diego, California. 125 pp.

6.0 APPENDICES

6.1 APPENDIX A. LISTING OF NORMIX SUBROUTINES

```
@ELT,L NORMIX.MAIN/CLUSTER
ELT 8R1 S7401C 11/07/80 11:23:19 (37)
  1.          33     C     PROGRAM MAIN                                                00000010
  2.          33     C                      NORMIX                                     00000020
  3.          33     C        A COPY OF THIS PROGRAM CAN BE OBTAINED BY SENDING THE AUTHOR* 00000030
  4.          33     C      A BLANK MAGNETIC TAPE.  TWO TAPE FORMATS ARE AVAILABLE AS FOLLOWS 00000040
  5.          33     C        9-TRACK                      7-TRACK                      00000050
  6.          33     C        EBCDIC                       BCD                          00000060
  7.          33     C        800 BPI                      556 BPI                      00000070
  8.          33     C        25 CARDS PER BLOCK           UNBLOCKED                    00000080
  9.          33     C     *THE AUTHOR'S NAME AND ADDRESS ARE GIVEN BELOW IN A6, LINE 17-19. 00000090
 10.          33     C...............................................................00000100
 11.          33     C   A.  IDENTIFICATION                                            00000110
 12.          33     C      1. TITLE= CLUSTER AND PATTERN ANALYSIS OF NORMAL MIXTURES  00000120
 13.          33     C      2. IDENTIFICATION= C7 NORMIX                               00000130
 14.          33     C      3. CATEGORY= MULTIVARIATE STATISTICS                       00000140
 15.          33     C      4. COMPILER/OPERATING SYSTEM= FORTRAN IV,G,OS-360/65       00000150
 16.          33     C      5. DATE= JUNE 1971                                         00000160
 17.          33     C      6. PROGRAMMER= JOHN H. WOLFE                               00000170
 18.          33     C                    U.S.N. PERSONNEL AND TRAINING RESEARCH LABORATORY 00000180
 19.          33     C                    SAN DIEGO, CALIFORNIA 92152                  00000190
 20.          33     C                                                                 00000200
 21.          33     C   B.  PURPOSE                                                   00000210
 22.          33     C       THIS PROGRAM SEEKS MAXIMUM LIKELIHOOD ESTIMATES OF THE    00000220
 23.          33     C   PARAMETERS OF A MIXTURE OF MULTIVARIATE NORMAL DISTRIBUTIONS. 00000230
 24.          33     C   AN OPTION PERMITS THE USER TO SPECIFY WHETHER THE COVARIANCE  00000240
 25.          33     C   MATRICES WITHIN TYPES WILL BE THE SAME OR DIFFERENT.         00000250
 26.          33     C                                                                 00000260
 27.          33     C       REFERENCE.                                                00000270
 28.          33     C        WOLFE, JOHN H.                                           00000280
 29.          33     C          PATTERN CLUSTERING BY MULTIVARIATE MIXTURE ANALYSIS.   00000290
 30.          33     C          MULTIVARIATE BEHAVIORAL RESEARCH, 1970,5,329-350.     00000300
 31.          33     C                                                                 00000310
 32.          33     C   C.  USAGE                                                     00000320
 33.          33     C                                                                 00000330
 34.          33     C      1. STORAGE REQUIREMENTS= VARIABLE, DEPENDING ON THE DATA.  00000340
 35.          33     C        THE ARRAYS ARE DIMENSIONED AT EXECUTION TIME. A MINIMAL  00000350
 36.          33     C        PROBLEM WOULD REQUIRE 120000 BYTES OF STORAGE.           00000360
 37.          33     C        THE FIRST TWO LINES OF THE PRINTOUT GIVE THE STORAGE REQUIRED 00000370
 38.          33     C        FOR THE PARTICULAR PROBLEM.                              00000380
 39.          33     C                                                                 00000390
 40.          33     C      2. RESTRICTIONS=  THE NUMBER OF TYPES MUST NOT EXCEED 20.  THERE 00000400
 41.          33     C        ARE NO FIXED LIMITS ON THE NUMBER OF VARIABLES OR SAMPLE SIZE.00000410
 42.          33     C        HOWEVER, ALL DATA AND ARRAYS MUST FIT INTO CORE.         00000420
 43.          33     C                                                                 00000430
 44.          33     C      3. ERROR MESSAGES= BAD INITIAL ESTIMATES OR DIVERGING ITERATIONS 00000440
 45.          33     C        CAN CAUSE STRANGE ESTIMATES OF THE PARAMETERS- SUCH AS   00000450
 46.          33     C        NEGATIVE VARIANCES OR SINGULAR CORRELATION MATRICES WHICH 00000460
 47.          33     C        RESULT IN SYSTEM DIAGNOSTICS. IN SUCH A CASE, THE USER MAY 00000470
 48.          33     C        HAVE TO SPECIFY HIS OWN INITIAL ESTIMATES ON THE INPUT-FORM, 00000480
 49.          33     C        OR RESCALE THE DATA TO ELIMINATE DICHOTOMOUS VARIABLES.  00000490
 50.          33     C      4. TIME=                                                   00000500
 51.          33     C        ON THE IBM 360/65, THE CPU TIME IN MINUTES IS GIVEN BY   00000510
 52.          33     C        THE FOLLOWING TWO FORMULAS                               00000520
 53.          33     C        COMMON COVARIANCE MATRIX OPTION                          00000530
 54.          33     C          MINUTES =(1.0E-6)*( 1067*(2*T-7) +M*(2.865*N*V +4.961*M) 00000540
 55.          33     C                              +4.25*T*V**2     +2.12*T*V**3 +67*N*T 00000550
33      C                        +1.73*N*T*T*(V+2)    )        .         00000560
33      C     DIFFERENT COVARIANCE MATRICES OPTION                       00000570
33      C       MINUTES =(1.0E-6)*( M*(2.865*N*V +4.961*M) +4.25*T*V**2  00000580
33      C                          +67*N*T +0.62*N*I*T*(V+2)  )      .   00000590
33      C       WHERE,                                                   00000600
33      C         T = NUMBER OF TYPES +1                                 00000610
33      C         V = NUMBER OF VARIABLES                                00000620
33      C         I = NUMBER OF ITERATIONS (USUALLY 40-80 )              00000630
33      C         M = NUMBER OF KMEANS                                   00000640
33      C         N = SAMPLE SIZE                                        00000650
33      C   5. TAPES=                                                    00000660
33      C     THE DATA RESIDE ON THE LOGICAL UNIT SPECIFIED BY THE USER  00000670
33      C     ON THE FIFTH INPUT-FORM CARD. THIS LOGICAL UNIT MUST NOT BE 00000680
33      C     UNIT 2,3,4,6, OR 7.  UNITS 2 AND 3 ARE UNFORMATTED SCRATCH 00000690
33      C     TAPES OR DISK AREAS. UNIT 4  CONTAINS CARD-IMAGE OUPUT OF  00000700
33      C     DISCRIMINANT SCORES FOR EACH INDIVIDUAL. UNITS 6 AND 7 ARE 00000710
33      C     THE STANDARD SYSTEM PRINT AND PUNCH UNITS, RESPECTIVELY.   00000720
33      C     THE INPUT-FORM CARDS AND INITIAL ESTIMATES ARE READ FROM   00000730
33      C     THE SYSTEM INPUT UNIT 5.                                   00000740
33      C   6. INPUT                                                     00000750
33      C                                                                00001760
33      C    **********SETUP FOR INITIAL ESTIMATE CARDS********************00001770
33      C                                                                00001780
33      C    CARD 1A   PROPORTION OF POPULATION IN TYPE 1      F8.8 FORMAT  00001790
33      C    CARD 1B   MEANS OF VARIABLES IN TYPE 1            10F8.4 FORMAT00001800
33      C                                                                00001810
33      C    CARD 2A   PROPORTION OF POPULATION IN TYPE 2      F8.8 FORMAT  00001820
33      C    CARD 2B   MEANS OF VARIABLES IN TYPE 2            10F8.4 FORMAT00001830
33      C    ...       ...       ...                ...        ...          00001840
33      C    ...       ...       ...                ...        ...          00001850
33      C    CARD RA   PROPORTION OF POPULATION IN LAST TYPE   F8.8 FORMAT  00001860
33      C    CARD RB   MEANS OF VARIABLES IN LAST TYPE         10F8.4 FORMAT00001870
33      C                                                                00001880
33      C    CARD  C   STANDARD DEVIATIONS WITHIN TYPE         10F8.4 FORMAT00001890
33      C                                                                00001900
33      C    CARD  D-1 FIRST ROW OF CORRELATIONS WITHIN TYPE   10F8.4 FORMAT00001910
33      C    CARD  D-2 SECOND ROW OF CORRELATIONS WITHIN TYPE  10F8.4 FORMAT00001920
33      C    ...       ...       ...                ...        ...          00001930
33      C    CARD  D-M LAST  ROW OF CORRELATIONS WITHIN TYPE   10F8.4 FORMAT00001940
33      C                                                                00001950
33      C    IF THE OPTION FOR DIFFERENT COVARIANCE MATRICES IS SELECTED    00001960
33      C    THEN A SET OF STANDARD DEVIATIONS AND CORRELATION CARDS
33      C    (C-D) FOLLOWS THE MEANS CARD FOR EACH TYPE.
33      C                                                                00001990
33      C                                                                00002000
33      C   7. OUTPUT                                                    00002010
33      C                                                                00002020
33      C     --HIERARCHICAL GROUPING---                                 00002030
33      C          THIS ROUTINE PRINTS THREE ITERATIONS AS FOLLOWS:      00002040
33      C       1) MAHALANOBIS DISTANCES FROM TOTAL COVARIANCE MATRIX.   00002050
33      C       2) MAHALANOBIS DISTANCES FROM WITHIN-GROUP COVARIANCES   00002060
33      C          OF 10 GROUPS FOUND IN STEP 1.                         00002070
33      C       3) MAHALANOBIS DISTANCES FROM WITHIN-GROUP COVARIANCES   00002080
33      C          OF 10 GROUPS FOUND IN STEP 2.                         00002090
33      C                                                                00002100
33      C        RANK= LINE NUMBER OF THE PRINTOUT                       00002110
33      C        ITEM= INDIVIDUAL OR OBJECT BEING CLUSTERED              00002120
```

```
33   C         KMEAN= THE KMEAN CLUSTER MEMBERSHIP OF THE ITEM                 00002130
33   C                THE FIRST ITEM OF EACH KMEAN BEGINS WITH A - SIGN.       00002140
33   C         STAGE= THE NUMBER OF GROUPS REMAINING AFTER THIS CLUSTER        00002150
33   C                WAS MERGED WITH THE PRECEDING CLUSTER.                   00002160
33   C         THE RIGHT HAND SIDE OF THE PAGE IS READ COLUMNWISE. EACH        00002170
33   C         COLUMN REPRESENTS A STAGE IN THE HIERARCHICAL GROUPING.         00002180
33   C         EACH GROUP BEGINS WITH A PERIOD AND CONTINUES WITH AN X         00002190
33   C                                                                         00002200
33   C       --ITERATION 0 ---                                                 00002210
33   C         THIS PRINTOUT GIVES THE INITIAL ESTIMATES USED TO BEGIN THE    00002220
33   C         ITERATIVE SOLUTION TO THE LIKELIHOOD EQUATIONS. THESE           00002230
33   C         ESTIMATES WILL BE THE SAME  AS THE CORRESPONDING                00002240
33   C         HIERARCHICAL GROUPING UNLESS THE USER INCLUDES HIS OWN          00002250
33   C         INITIAL ESTIMATE CARDS.                                         00002260
33   C                                                                         00002270
33   C       --ITERATION-LAST--                                                00002280
33   C         GIVES THE CONVERGED SOLUTION TO THE LIKELIHOOD EQUATIONS.       00002290
33   C                                                                         00002300
33   C       --DISCRIMINANT FUNCTIONS---                                       00002310
33   C         GIVES THE WEIGHTS TO APPLY TO THE RAW SCORES SO THAT            00002320
33   C         DISCRIMINANT SCORES WILL HAVE MAXIMAL DISCRIMINATION BETWEEN00002330
33   C         GROUPS AND THE IDENTITY MATRIX FOR THE WITHIN-GROUP             00002340
33   C         COVARIANCES.                                                    00002350
33   C                                                                         00002360
33   C       --PROBABILITIES OF TYPE MEMBERSHIP--(SELF-EXPLANATORY)            00002370
33   C                                                                         00002380
33   C       --PRINTER PLOT---                                                 00002390
33   C         EACH POINT HAS A NUMBER OR LETTER DESIGNATING THE TYPE FOR      00002400
33   C         WHICH THIS INDIVIDUAL HAS THE HIGHEST PROBABILITY OF            00002410
33   C         MEMBERSHIP. THE FIRST 9 CLUSTERS ARE IDENTIFIED BY THE          00002420
33   C         DIGITS 1-9. THE NEXT 11 CLUSTERS ARE IDENTIFIED BY THE          00002430
33   C         LETTERS A-K. AN * INDICATES A PLACE WHERE THERE ARE TWO         00002440
33   C         INDIVIDUALS WITH DIFFERENT CLUSTER MEMBERSHIP. POINTS WHICH    00002450
33   C         LIE BEYOND THE BOUNDARIES OF THE GRAPH ARE PROJECTED AND        00002460
33   C         PLOTTED AT THE BOUNDARIES.                                      00002470
33   C                                                                         00002480
33   C     8.  ASSISTANCE
33   C    FOR ASSISTANCE IN RUNNING THE 1979 UNIVAC 1100 VERSION OF
33   C    THIS PROGRAM, PLEASE CONTACT HAROLD CRUTCHER,
33   C    ASHEVILLE, NC (704-253-2539), OR MAURICE GRAVES,
33   C    CHAPEL HILL, NC (919-942-3041).
33   C.........................................................................00002490
33   C    THIS 1970 VERSION OF NORMIX FOR THE IBM360 COMBINES THE OLD          00002500
33   C    NORMAP AND NORMIX PROGRAMS FOR THE 1604 INTO A SINGLE PROGRAM        00002510
33   C                                                                         00002520
33   C    THIS (MAIN) PROGRAM DIMENSIONS THE ARRAYS.                           00002530
33   C                                                                         00002540
33        IMPLICIT REAL*8 (A-H,O-Z)                                            00002550
33        REAL*8 LAMBDA,MEAN,MU,LIKLY
33        LOGICAL NORMAP                                                       00002580
34        DIMENSION MSG(15)                                                     00002590
33        DIMENSION A(15000)                                                   00002610
33        COMMON/A/LIKLY,XLIMIT,NVBLES,NSAMPL,NTYPES,NSEMI,NPARAM,NORMAP,KX 00002620
33        COMMON/B/CONVRG,CRITRN,RATIO,HTEST,OLDLIK,EN,BIGEST,THRESH,GUARD, 00002630
33       1ENMIN,TITLE(18),FMT( 8),    MAXTYP,MAXAV,NCLUST,KSCRAT,INP,KPUN,KV,00002640
33       2 LOT,LACK,LUCK,IPRINT,LTIME,MAXT,ITER,ITERX,LTYE,ITMC,NRANK,IVALU 00002650
33       3,METHOD,NTYP(20)
```

```
35           DATA MSG/'    ','CORE',' SIZ','E TO','O SM','ALL.',' RE','DUCE',
37          *' NVB','LES',',' NSA','MPL',',' OR ',' MAXA','V. '/
33        10 CALL INFORM                                                   00002670
33           IF (NVBLES.EQ.-1) GO TO 999                                   00002680
33           IF (NSAMPL.LT.10) GO TO 10                                    00002690
33           REWIND 2
33           REWIND 3
33           REWIND 4
33           REWIND 8
33           NB = 0                                                        00002740
33           B = 0.                                                        00002750
33           WRITE (8,50) NB,NB,NB,B                                       00002760
33        50 FORMAT (3A4,A8)                                               00002770
33           ENDFILE 8                                                     00002780
33           REWIND 8                                                      00002790
33     C     CALCULATE NUMBER OF BYTES OCCUPIED BY EACH ARRAY              00002800
33           NB=MAX0(MAXAV*8,NVBLES*8,168)                                 00002810
33           NB=((NB+7)/8)*8                                               00002820
33           NC=MAXTYP*8                                                   00002830
33           NX=NVBLES*NSAMPL*8                                            00002840
33           NX=MAX0(NX,24000,MAXAV*(MAXAV+1)*4 )                          00002850
33           NX=((NX+7)/8)*8                                               00002860
33           NVC=NVBLES*NC                                                 00002870
33           NV=NVBLES*8                                                   00002880
33           NP=MAX0(NSEMI*8,NPARAM*NC)                                    00002890
33           IF(NORMAP) GO TO 2                                            00002900
33           NCOV=NSEMI*NC                                                 00002910
33           NVT=8                                                         00002920
33           NR=8                                                          00002930
33           NVV=NV                                                        00002940
33           GO TO 3                                                       00002950
33         2 NCOV=NSEMI*24                                                 00002960
33           NVT=NVBLES*NV                                                 00002970
33           NR=NV                                                         00002980
33           NVV=NVC                                                       00002990
33         3 N1=1                                                          00003000
33           IN1=LOC(A(N1))
33           N2=N1+NC/8                                                    00003010
33           IN2=LOC(A(N2))
33           N3=N2+NVC/8                                                   00003020
33           IN3=LOC(A(N3))
33           N4=N3+NCOV/8                                                  00003030
33           IN4=LOC(A(N4))
33           N5=N4+NP/8                                                    00003040
33           IN5=LOC(A(N5))
33           N6=N5+NC/8                                                    00003050
33           IN6=LOC(A(N6))
33           N7=N6+NVC/8                                                   00003060
33           IN7=LOC(A(N7))
33           N8=N7+NVV/8                                                   00003070
33           IN8=LOC(A(N8))
33           N9=N8+NVBLES**2                                               00003080
33           IN9=LOC(A(N9))
33           N10=N9+NVT/8                                                  00003090
33           IN10=LOC(A(N10))
33           N11=N10+NVT/8                                                 00003100
33           IN11=LOC(A(N11))
```

```
  227.         33           N12=N11+NB/8                                                        00003110
  228.         33           IN12=LOC(A(N12))
  229.         33           N13=N12+NB/8                                                        00003120
  230.         33           IN13=LOC(A(N13))
  231.         33           N14=N13+NB/8                                                        00003130
  232.         33           IN14=LOC(A(N14))
  233.         33           N15=N14+NP/8                                                        00003140
  234.         33           IN15=LOC(A(N15))
  235.         33           N16=N15+MAXAV+NVBLES                                                00003150
  236.         33           IN16=LOC(A(N16))
  237.         33           N17=N16+MAXAV                                                       00003160
  238.         33           TN17=LOC(A(N17))
  239.         33           N18=N17+NB/8                                                        00003170
  240.         33           IN18=LOC(A(N18))
  241.         33           N19=N18+NX/8                                                        00003180
  242.         33           IN19=LOC(A(N19))
  243.         33           N20 = N19+NSAMPL                                                    00003190
  244.         33           IN20=LOC(A(N20))
  245.         33           PRINT 8,N20                                                         00003200
  246.         33           PRINT 9,N1,IN1,N2,IN2,N3,TN3,N4,IN4,N5,IN5,N6,IN6,N7,IN7,
  247.         33          1N8,IN8,N9,IN9,N10,IN10,N11,IN11,N12,IN12,N13,IN13,N14,IN14,
  248.         33          2N15,IN15,N16,IN16,N17,IN17,N18,IN18,N19,TN19,N20,TN20
  249.         33         9 FORMAT(////,T20,'ARRAY',T40,'LOCATION',/,20(/,T20,
  250.         33          1 'A(',I5,')',T40,O8))
  251.         33           IF (N20.LE.15000) GO TO 6                                           00003210
  252.         37           CALL PUTMSG (MSG,58)                                                00003220
  253.         33           STOP                                                                00003230
  254.         33      C                                                                        00003240
  255.         33      C    CALL GETCOR(NORMIX,NC,NVC,NCOV,NP,NC,NVC,NVV,8*NVBLES**2,NVT,NVT  00003250
  256.         33      C   1,NB,NB,NB,NB,MAXAV*NVBLES*4,MAXAV*4,NB,NX,NSAMPL*4)                00003260
  257.         33      C                                                                        00003270
  258.         33      C       GETCOR IS AN ASSEMBLY-LANGUAGE ROUTINE WHICH RESERVES STORAGE    00003280
  259.         33      C    SPACE FOR THE ARRAYS AT EXECUTION TIME. AN ALTERNATIVE TO USING     00003290
  260.         33      C    GETCOR IS TO RECOMPILE THTS (MATN) PROGRAM, REPLACING THE           00003300
  261.         33      C      ' EXTERNAL NORMIX'  CARD WITH                                     00003310
  262.         33      C    DIMENSION A(10000)                                                  00003320
  263.         33      C    OR LARGER DIMENSION IF NECESSARY. THEN REPLACE THE ' CALL GETCOR '00003330
  264.         33      C    CARDS WITH THE FOLLOWING THREE CARDS:                               00003340
  265.         33         6  CALL NORMIX(A,A(N2),A(N3),A(N4),A(N5),A(N6),A(N7),A(N8),A(N9)    00003350
  266.         33          1,A(N10),A(N11),A(N12),A(N13),A(N14),A(N15),A(N16),A(N17),A(N18)  00003360
  267.         33          2,A(N19))                                                          00003370
  268.         33      C                                                                        00003380
  269.         33           N2 = N1+(NSAMPL+1)/2                                                00003390
  270.         33           N3 = N2+(NSAMPL+1)/2                                                00003400
  271.         33           CALL PRTTYP (A,A(N2),A(N3))                                         00003410
  272.         33     999   STOP                                                                00003420
  273.         33      C                                                                        00003430
  274.         33         8  FORMAT (//,' ARRAY A REQUIRES',I6,' LOCATIONS')                    00003440
  275.         33           END                                                                 00003450

END ELT.  ERRORS: NONE. TIME:  2.290 SEC. IMAGE COUNT: 275


@FIN
```

```
000251     027           CALL PUTMSG (MSG,30)                                               00003220
000252     027           STOP                                                               00003230
000253     027     C                                                                        00003240
000254     027     C     CALL GETCOR(NORMIX,NC,NVC,NCOV,NP,NC,NVC,NVV,8*NVBLES**2,NVT,NVT   00003250
000255     027     C    1,NB,NB,NB,NB,MAXAV*NVBLES*4,MAXAV*4,NB,NX,NSAMPL*4)                00003260
000256     027     C                                                                        00003270
000257     027     C        GETCOR IS AN ASSEMBLY-LANGUAGE ROUTINE WHICH RESERVES STORAGE   00003280
000258     027     C     SPACE FOR THE ARRAYS AT EXECUTION TIME. AN ALTERNATIVE TO USING    00003290
000259     027     C     GETCOR IS TO RECOMPILE THIS (MAIN) PROGRAM, REPLACING THE          00003300
000260     027     C       ' EXTERNAL NORMIX'  CARD WITH                                    00003310
000261     027     C     DIMENSION A(10000)                                                 00003320
000262     027     C     OR LARGER DIMENSION IF NECESSARY. THEN REPLACE THE ' CALL GETCOR ' 00003330
000263     027     C     CARDS WITH THE FOLLOWING THREE CARDS:                              00003340
000264     027         6 CALL NORMIX(A,A(N2),A(N3),A(N4),A(N5),A(N6),A(N7),A(N8),A(N9)      00003350
000265     027          1,A(N10),A(N11),A(N12),A(N13),A(N14),A(N15),A(N16),A(N17),A(N18)    00003360
000266     027          2,A(N19))                                                           00003370
000267     027     C                                                                        00003380
000268     027           N2 = N1+(NSAMPL+1)/2                                               00003390
000269     027           N3 = N2+(NSAMPL+1)/2                                               00003400
000270     027           CALL PRTTYP (A,A(N2),A(N3))                                        00003410
000271     027       999 STOP                                                               00003420
000272     027     C                                                                        00003430
000273     027         8 FORMAT (//,' ARRAY A REQUIRES',I6,' LOCATIONS')                    00003440
000274     027           END                                                                00003450

END ELT.

@FIN
```

```
@ELT,L N.AITKEN/CLUSTER
EL1007 SL73R1 02/25/80 16:08:53 (28,)
000001       024          SUBROUTINE AITKEN (LAMBDA,MEAN,COVIN,X,R,S,B,V,G,MU,OLAM,OMEAN)    00000010
000002       024    C     THIS SUBROUTINE DOES ALL THE COMPUTATIONS OF A SINGLE ITERATION    00000030
000003       024    C     FOR IMPROVED ESTIMATES OF THE PARAMETERS.                          00000040
000004       024          IMPLICIT REAL*8 (A-H,O-Z)                                          00000050
000005       024          REAL*8 LAMBDA,MEAN,MU,LIKLY                                        00000060
000006       024          LOGICAL NORMAP                                                     00000080
000007       024          COMMON/A/LIKLY,XLIMIT,NVBLES,NSAMPL,NTYPES,NSEMI,NPARAM,NORMAP,KX 00000090
000008       024          COMMON/B/CONVRG,CRITRN,RATIO,HTEST,OLDLIK,EN,BIGEST,THRESH,GUARD, 00000100
000009       024         1ENMIN,TITLE(18),FMT( 8),   MAXTYP,MAXAV,NCLUST,KSCRAT,INP,KPUN,KV,00000110
000010       024         2 LOT,LACK,LUCK,IPRINT,LTIME,MAXT,ITER,ITERX,LTYE,ITMC,NRANK,IVALUE00000120
000011       028         3,METHOD,NTYP(20)
000012       024          DIMENSION LAMBDA(1),MEAN(NVBLES,NTYPES),COVIN(NSEMI,NTYPES),       00000140
000013       024         1 MU(NPARAM,NTYPES),B(1),OLAM(1),OMEAN(NVBLES,NTYPES),              00000150
000014       024         2R(1),S(1),G(1 ), X(1),V(NVBLES,NTYPES)                             00000160
000015       024          XLIMIT=AMINO(40,(ITER/5)*10)                                       00000170
000016       024    C     WHEN XLIMIT=0, ALL POINTS ARE ASSIGNED A MEMBERSHIP PROBABILITY OF00000180
000017       024    C     1.0 IN THEIR NEAREST CLUSTER.                                      00000190
000018       024          N=NVBLES-1                                                         00000200
000019       024          LUCK=1                                                             00000210
000020       024          BIGEST=20.0                                                        00000220
000021       024          SMALL=1.0-1.0/BIGEST                                               00000230
000022       024          LQQ=NSEMI                                                          00000240
000023       024          DO 3 L=2,NTYPES                                                    00000250
000024       024          IF(L.GT.2.AND.NORMAP) GO TO 2                                      00000260
000025       024          CALL SYMINV (COVIN,NVBLES,DETM,R,S,G,LQQ)                          00000270
000026       024    C   CHECK IF ZERO PIVOT                                                  00000280
000027       024          DO 30 LT = 1,NVBLES                                                00000290
000028       024          IF (G(LT).EQ.1.) GO TO 90                                          00000300
000029       024       30 CONTINUE                                                           00000310
000030       024          IF(NORMAP) GO TO 2                                                 00000320
000031       024          IF (ENMIN.LT.EN*LAMBDA(L)) GO TO 2                                 00000330
000032       024          LSING=L-1                                                          00000340
000033       024          NSING=EN*LAMBDA(L)                                                 00000350
000034       024          OLDLIK=-1.0D+10                                                    00000360
000035       024          PRINT 101,LSING,NSING,OLDLIK                                       00000370
000036       024      101 FORMAT(' TYPE' I3,' WITH 'I3,' POINTS APPROACHING SINGULARITY'/   00000380
000037       024         1' LIKELIHOOD RE-INITIALIZED TO' D15.8)                             00000390
000038       024          IJ=0                                                               00000400
000039       024          DO 1 I=1,NVBLES                                                    00000410
000040       024          DO 1 J=I,NVBLES                                                    00000420
000041       024          IJ=IJ+1                                                            00000430
000042       024          COVIN(IJ,L)=COVIN(IJ,1)+MEAN(I,1)*MEAN(J,1)                        00000440
000043       024          DO 1 M=2,NTYPES                                                    00000450
000044       024    1     COVIN(IJ,L)=COVIN(IJ,L)-MU(I+1,M)*MU(J+1,M)/(MU(1,M)*EN)          00000460
000045       024          CALL SYMINV (COVIN,NVBLES,DETM,R,S.G,LQQ)                          00000470
000046       024    C   CHECK IF ZERO PIVOT                                                  00000480
000047       024          DO 31 LT = 1,NVBLES                                                00000490
000048       024          IF (G(LT).EQ.1.) GO TO 90                                          00000500
```

```
000049     024     31  CONTINUE                                                              00000510
000050     024    2       B(L)=DLOG(LAMBDA(L))-0.5*DETM                                      00000520
000051     024    3      LQQ=LQQ+NSEMI                                                       00000530
000052     024           LIKLY=0.0                                                           00000540
000053     024           DO 4 L=2,NTYPES                                                     00000550
000054     024           DO 4 I=1,NPARAM                                                     00000560
000055     024    4      MU(I,L)=0.0                                                         00000570
000056     024           IF(NORMAP) GO TO 40                                                 00000580
000057     024           III=NVBLES                                                          00000590
000058     024           II=1                                                                00000600
000059     024           DO 6 I=1,NVBLES                                                     00000610
000060     024           DO 5 L=2,NTYPES                                                     00000620
000061     024    5      COVIN(II,L)=COVIN(II,L)/2.0                                         00000630
000062     024           II=II+III                                                           00000640
000063     024    6      III=III-1                                                           00000650
000064     024           GO TO 47                                                            00000660
000065     024    40     K=1                                                                 00000670
000066     024           DO 42 I=1,NVBLES                                                    00000680
000067     024           LIKLY=LIKLY-0.5*COVIN(K,2)*(COVIN(K,1)+MEAN(I,1)**2)*EN             00000690
000068     024           K=K+1                                                               00000700
000069     024           IF (I.GT.N) GO TO 42                                                00000710
000070     024           DO 41 J=I,N                                                         00000720
000071     024           LIKLY=LIKLY-COVIN(K,2)*(COVIN(K,1)+MEAN(J+1,1)*MEAN(I,1))*EN        00000730
000072     024    41     K=K+1                                                               00000740
000073     024    42     CONTINUE                                                            00000750
000074     024           DO 46 L=2,NTYPES                                                    00000760
000075     024           DO 45 I=1,NVBLES                                                    00000770
000076     024           K=I                                                                 00000780
000077     024           N=NVBLES                                                            00000790
000078     024           V(I,L)=0.0D0                                                        00000800
000079     024           IF (I.EQ.1) GO TO 44                                                00000810
000080     024           DO 43 J=2,I                                                         00000820
000081     024           V(I,L)=V(I,L)+COVIN(K,2)*MEAN(J-1,L)                                00000830
000082     024           N=N-1                                                               00000840
000083     024           K=N+K                                                               00000850
000084     024    43     CONTINUE                                                            00000860
000085     024    44     DO 45 J=I,NVBLES                                                    00000870
000086     024           V(I,L)=V(I,L)+COVIN(K,2)*MEAN(J   ,L)                               00000880
000087     024    45     K=K+1                                                               00000890
000088     024           DO 46 I=1,NVBLES                                                    00000900
000089     024    46     B(L)=B(L)-V(I,L)*MEAN(I,L)*0.5D0                                    00000910
000090     024     47    CALL MOMENT (MEAN,COVIN,X,R,S,B,G,MU,V)                             00000920
000091     024           BYG=1.0                                                             00000930
000092     024           IF (METHOD.GT.0) GO TO 8                                            00000940
000093     024           BYG=BIGEST                                                          00000950
000094     024           DO 7 L=2,NTYPES                                                     00000960
000095     024           DELTA=MU(1,L)/EN-LAMBDA(L)                                          00000970
000096     024           BIGA=LAMBDA(L)-OLAM(L)                                              00000980
000097     024           SML=1.0-1.0/BYG                                                     00000990
000098     024           IF (DABS(DELTA).LT.SML*DABS(BIGA)) BYG=BIGA/(BIGA-DELTA)            00001000
```

```
000099   024          IF(DELTA) 21,7,22                                                       00001010
000100   024    21    BOUND=DMAX1(LAMBDA(L)*0.5D0,1.0D0/EN)                                   00001020
000101   024          IF(BOUND.LT.-DELTA*BYG) BYG=-BOUND/DELTA                                00001030
000102   024          GO TO 7                                                                 00001040
000103   024    22    BOUND=DMIN1((1.0D0-LAMBDA(L))*0.5D0,(EN-1.0D0)/EN)                      00001050
000104   024          IF(BOUND.LT.DELTA*BYG) BYG=BOUND/DELTA                                  00001060
000105   024    7     CONTINUE                                                                00001070
000106   024    8     DO 12 L=2,NTYPES                                                        00001080
000107   024          IF(MU(1,L).LT. 1.0D0) MU(1,L)=1.0D0                                     00001090
000108   024          OLAM(L)=LAMBDA(L)                                                       00001100
000109   024          LAMBDA(L)=LAMBDA(L)+BYG*(MU(1,L)/EN-LAMBDA(L))                          00001110
000110   024          LAMBDA(L)=DMAX1(1.0D-1/EN,LAMBDA(L))                                    00001120
000111   024          IJ=0                                                                    00001130
000112   024          DO 9 I=1,NVBLES                                                         00001140
000113   024          BIGA=MEAN(I,L)-OMEAN(I,L)                                               00001150
000114   024          BIGC=MU(I+1,L)/MU(1,L)-MEAN(I,L)                                        00001160
000115   024          BIG=BIGEST                                                              00001170
000116   024          IF (DABS(BIGC).LT.SMALL*DABS(BIGA)) BIG=BIGA/(BIGA-BIGC)                00001180
000117   024          IF (METHOD.GT.0) BIG=1.0                                                00001190
000118   024          OMEAN(I,L)=MEAN(I,L)                                                    00001200
000119   024    9     MEAN(I,L)=MEAN(I,L)+BIG*BIGC                                            00001210
000120   024          IF(NORMAP) GO TO 12                                                     00001220
000121   024          KJ=NVBLES+1                                                             00001230
000122   024          DO 10 I=1,NVBLES                                                        00001240
000123   024          DO 10 J=I,NVBLES                                                        00001250
000124   024          IJ=IJ+1                                                                 00001260
000125   024          KJ=KJ+1                                                                 00001270
000126   024    10    COVIN(IJ,L)=(MU(KJ,L)-MU(I+1,L)*MEAN(J,L)-MU(J+1,L)*MEAN(I,L))/MU(00001280
000127   024         11,L)+MEAN(I,L)*MEAN(J,L)                                                00001290
000128   024    12    CONTINUE                                                                00001300
000129   024          METHOD=IABS(METHOD-1)                                                   00001310
000130   024          IF (LIKLY.LT.OLDLIK) METHOD=LUCK                                        00001320
000131   024          IF(.NOT.NORMAP) RETURN                                                  00001330
000132   024          DO 11 I=1,NVBLES                                                        00001340
000133   024          DO 11 J=I,NVBLES                                                        00001350
000134   024          IJ=IJ+1                                                                 00001360
000135   024          COVIN(IJ,2)=COVIN(IJ,1)+MEAN(I,1)*MEAN(J,1)                             00001370
000136   024          DO 11 L=2,NTYPES                                                        00001380
000137   024    11    COVIN(IJ,2)=COVIN(IJ,2)-MU(I+1,L)*MU(J+1,L)/(MU(1,L)*EN)                00001390
000138   024       90 RETURN                                                                  00001400
000139   024          END                                                                     00001410
000140   024          FUNCTION AREA (W)                                                       00001420
000141   024    C     AREA ABOVE W ON NORMAL CURVE                                            00001430
000142   024          IMPLICIT REAL*8 (A-H,O-Z)                                               00001440
000143   024          X=W*.7071067811865475244O                                               00001450
000144   024          KOT=2                                                                   00001460
000145   024          IF (X) 1,1,2                                                            00001470
000146   024    1     KOT=1                                                                   00001480
000147   024          X=-X                                                                    00001490
000148   024    2     AREA=0.5/((((((.0000430638*X+.0002765672)*X+.0001520143)*X+.009270000001500
000149   024         15272)*X+.0422820123)*X+.0705230784)*X+1.0)**16                          00001510
```

```
0CC150        024          GO TO (3,4), KOT                                                    00001520
0CC151        024    3     AREA=1.0-AREA                                                       00001530
00C152        024    4     RETURN                                                              00001540
000153        124          END                                                                 00001550

END ELT.
```

```
@ELT,L NORMIX.ASSIGN/CLUSTER
ELT 8R1 S74Q1C 11/04/80 12:42:35 (34)
   1.        30           SUBROUTINE ASSIGN(LAMBDA,MEAN,COVIN,X,R,S,B,V,G,MU,T,W,BMAT)       00000010
   2.        30       C   THIS ROUTINE PRINTS THE PROBABILITIES OF MEMBERSHIP OF EACH        00000030
   3.        30       C   INDIVIDUAL IN EACH TYPE.                                           00000040
   4.        30       C   IT COMPUTES AND PRINTS DISCRIMINANT FUNCTIONS AND EACH INDIVIDUALS00000050
   5.        30       C   DISCRIMINANT SCORES.                                               00000060
   6.        30       C    DISCRIMINANT ANALYSIS.  SEE THE FOLLOWING REFERENCES              00000070
   7.        30       C   COOLEY+LOHNES. MULTIVARIATE PROCED.BEHAV.SCIENCES  PAGES 117,189. 00000080
   8.        30       C   FRIEDMAN AND RUBIN. JOURN.AM. STAT. ASSN. 1967, PAGE 1167.         00000090
   9.        30       C                                                                      00000100
  10.        30           IMPLICIT REAL*8 (A-H,O-Z)                                          00000110
  11.        30           REAL*8 LAMBDA,MEAN,MU,LIKLY                                        00000120
  12.        30           REAL*4 CELLS,COMP,CS,Q,WORK1,WORK2,Y,YBAR,SIGMA
  13.        30           LOGICAL NORMAP                                                     00000140
  14.        30           COMMON/A/LIKLY,XLIMIT,NVBLES,NSAMPL,NTYPES,NSEMI,NPARAM,NORMAP,KX 00000150
  15.        30           COMMON/B/CONVRG,CRITRN,RATIO,HTEST,OLDLIK,EN,BIGEST,THRESH,GUARD, 00000160
  16.        30          1ENMIN,TITLE(18),FMT( 8),   MAXTYP,MAXAV,NCLUST,KSCRAT,INP,KPUN,KV,00000170
  17.        30          2 LOT,LACK,LUCK,IPRINT,LTIME,MAXT,ITER,ITERX,LTYE,ITMC,NRANK,IVALUE00000180
  18.        30          3,METHOD,NTYP(20)
  19.        30           COMMON/D/THIER,MERGE,IGEN,IZQ,IHIS
  20.        30           COMMON/C/YBAR,SIGMA
  21.        30           COMMON/E/MTIME,NTIME,IOVER
  22.        30           COMMON/F/IMAP,IUNITS,IPROF,IBACK,IPOL(20),SCAL(20),POWER,IWIND(2),
  23.        30          ,NWIND
  24.        33           COMMON/H/SBACK(20,20)
  25.        30           DIMENSION LAMBDA(1),G(1),R(1),S(1),                                00000200
  26.        30          1  T(NVBLES,NVBLES),W(NVBLES,NVBLES),BMAT(NVBLES,NVBLES),          00000210
  27.        30          2 COVIN(NSEMI,NTYPES),X(NVBLES,NSAMPL),MU(NPARAM,NTYPES),          00000220
  28.        30          3 MEAN(NVBLES,NTYPES),B(1),V(1)                                                     00000230
  29.        30           DIMENSION Y(300),COMP(20),WORK1(110),WORK2(27),IGRP(300),
  30.        30          ,IWORK(50),CELLS(20),STS(20,4),XBACK(20,200)
  31.        30           EXTERNAL NORM
  32.        30           CALL ITIME(ITMDA)                                                  00000250
  33.        30           KC=3                                                               00000260
  34.        30           REWIND KC                                                          00000270
  35.        30           AL=LIKLY                                                           00000280
  36.        30           IF(NORMAP) GO TO 23                                                00000290
  37.        30           LQQ=NSEMI                                                          00000300
  38.        30           DO 32 L=2,NTYPES                                                   00000310
  39.        30           IJ=0                                                               00000320
  40.        30           DO 31 I=1,NVBLES                                                   00000330
  41.        30           DO 31 J=I,NVBLES                                                   00000340
  42.        30           IJ=IJ+1                                                            00000350
  43.        30       31   MU(IJ,L)=COVIN(I,J,L)                                             00000360
  44.        30           CALL SYMINV (COVIN,NVBLES,DETM,R,S,G,LQQ)                          00000370
  45.        30       C  CHECK FOR ZERO PIVOT                                                00000380
  46.        30           DO 30 LT = 1,NVBLES                                                00000390
  47.        30           IF (G(LT).EQ.1.) GO TO 99                                          00000400
  48.        30         30  CONTINUE                                                         00000410
  49.        30             B(L)=DLOG(LAMBDA(L))-0.5*DETM                                    00000420
  50.        30       32   LQQ=LQQ+NSEMI                                                     00000430
  51.        30       C THE FOLLOWING 7 LINES INSERTED FROM NORMIX BULLETIN RECD SEPT'74     00000440
  52.        30           ITI = NVBLES                                                       00000450
  53.        30           IT = 1                                                             00000460
  54.        30           DO 6 I = 1,NVBLES                                                  00000470
  55.        30           DO 5 L = 2,NTYPES                                                  00000480
```

```
30     5   COVIN(II,L) = COVIN(II,L)/2.0
30         II = II+ITI
30     6   III = III-1
30         GO TO 24
30    23   MRANK= MINO(NTYPES-2,NVBLES)
30         IJ=0
30         DO 1 I=1,NVBLES
30         DO 1 J=I,NVBLES
30         IJ=IJ+1
30         W(I,J)=COVIN(IJ,2)
30         BMAT(I,J)=COVIN(IJ,1)-COVIN(IJ,2)
30    1    BMAT(J,I)=BMAT(I,J)
30    C    BMAT IS THE BETWEEN-GROUP COVARIANCE MATRIX
30    C    W IS THE WITHIN-GROUP COVARIANCE MATRIX
30         CALL NROOT(NVBLES,BMAT,W,R,T)
30         CALL TYTLE
30         PRINT 14,(L,L=1,MRANK)
30         DO 9 I=1,NVBLES
30    9    PRINT 15, I,(T(I,L),L=1,MRANK)
30         PRINT 16, (R(L),L=1,MRANK)
30         CALL ITIME(ITMDB)
30         TMD=(ITMDB-ITMDA)/100.0
30         IF(MTIME.EQ.1)PRINT 27,TMD
30    27   FORMAT(//40X,'TIME TO COMPUTE DISCRIMINANT FUNCTIONS =' F8.2,
30        1' SECONDS.')
30    24   LIKLY=0.000
30         NTP=NTYPES-1
30         LAP=45
30         DO 13 K=1,NSAMPL
30         IF(LAP.LT.45) GO TO 10
30         LAP=0
30         CALL TYTLE
30         PRINT 17
30         IF(NORMAP) PRINT 171
30         PRINT 172
30         PRINT 172
30         IF(NORMAP) PRINT 173,(I,I=1,NRANK)
30         PRINT 18, (I,I=1,NTP)
30         PRINT 172
30    10   LAP=LAP+1
30         CALL DENSIT (MEAN,COVIN,X,R,S,B,V,G,K)
30         LA= MINO(NTYPES,11)
30    C    PRINTOUT PROBABILITIES OF MEMBERSHIP
30         PRINT 19, K,(G(I),I=2,LA)
30         XMAX=0.0
30         DO 11 L=2,NTYPES
30         IF (G(L).LT.XMAX) GO TO 11
30         LOT=L
30    C     LOT = THE MOST PROBABLE TYPE FOR INDIVIDUAL K.
30         XMAX=G(L)
30    11   CONTINUE
30         IGRP(K)=LOT
30           WRITE (8,50) NTP,LOT,K,XMAX
30      50   FORMAT (4I4)
30         IF(.NOT.NORMAP) GO TO 25
30         DO 12 L=1,NRANK
30         R(L)=0.0
```

```
30           DO 12 I=1,NVBLES                                                   00001050
30     12    R(L)=R(L)+T(I,L)*(X(I,K)-MEAN(I,L))                                  00001060
30     C     SAVE DISCRIMINANT SCORES ON TAPE, IN CARD-IMAGE FORMAT
30           WRITE(KC)K,LOT,(R(II),II=1,NRANK)                                  0000107 0
30           WRITE(KSCRAT,20)K,LOT,R                                            00001080
30     C     PRINT DISCRIMINANT SCORES ALONGSIDE PROBABILITIES OF MEMBERSHIP
30           PRINT 21,(R(II),II=1,NRANK)                                        00001100
30     25    IF (NTYPES.LE.11) GO TO 13                                         00001120
30           PRINT 22, (G(I),I=12,NTYPES)                                       00001130
30     13    CONTINUE                                                           00001140
30           IF(IMAP.NE.1)GO TO 37
30     C
30     C     STORE CLUSTER MEMBERSHIPS FOR MAPPING
30     C
30           INR=11
30           DO 101 K=1,NSAMPL
30           III=IGRP(K)-1
30     101   WRITE(INR,100)K,III
30     100   FORMAT(2I10)
30     37    IF(IBACK.EQ.0)GO TO 39
30     C
30     C     CLUSTER STATISTICS FOR ORIGINAL DATA AND UNEQUAL COVARIANCE
30     C     MATRICES
30     C
30           DTRAD=.0174533
30           DO 260 ID=2,NTYPES
30           DO 380 I=1,NVBLES
30           DO 380 J=1,4
30     380   STS(I,J)=0.
30           DO 290 IP=1,NVBLES
30           II=0
30           DO 280 J=1,4
30     280   IF(IP.NE.IWIND(2))STS(IP,J)=0.0
30           DO 211 K=1,NVBLES
30     211   IF(IP.EQ.IPOL(K))GO TO 212
30           GO TO 215
30     212   IF(POWER.NE.0.)GO TO 213
30           DO 270 J=1,NSAMPL
30           IF(IGRP(J).NE.ID)GO TO 270
30           II=II+1
30           XBACK(IP,II)=DEXP(X(IP,J))-SCAL(IP)
30     270   CONTINUE
30           GO TO 217
30     213   POWER=1./POWER
30           DO 216 J=1,NSAMPL
30           IF(IGRP(J).NE.ID)GO TO 216
30           II=II+1
30           XBACK(IP,II)=X(IP,J)**POWER-SCAL(IP)
30     216   CONTINUE
30           GO TO 217
30     215   IF(IP.EQ.IWIND(1))GO TO 219
30           IF(IP.EQ.IWIND(2))GO TO 290
30           DO 218 J=1,NSAMPL
30           IF(IGRP(J).NE.ID)GO TO 218
30           II=II+1
30           XBACK(IP,II)=X(IP,J)
30     218   CONTINUE
```

```
      30      217   DO 200 J=1,II
      30      200   STS(IP,1)=STS(IP,1)+XBACK(IP,J)
      30            STS(IP,1)=STS(IP,1)/FLOAT(II)
      30            IF(NORMAP)GO TO 290
      30            DO 201 J=1,II
      30            XDIF=XBACK(IP,J)-STS(IP,1)
      30      201   STS(IP,2)=STS(IP,2)+XDIF*XDIF
      30            STS(IP,2)=DSQRT(STS(IP,2)/FLOAT(II-1))
      30            STS(IP,3)=STS(IP,1)+2.*STS(IP,2)
      30            STS(IP,4)=STS(IP,1)-2.*STS(IP,2)
      30            GO TO 290
      30      C
      30      C     IF WIND COMPONENTS ARE TO BE BACKTRANSFORMED, THE VARIATES
      30      C     FOR WEST COMPONENT AND SOUTH COMPONENT MUST BE ADJACENT,
      30      C     WITH IWIND(1)=WEST COMPONENT VARIATE AND IWIND(2)=SOUTH
      30      C     COMPONENT VARIATE
      30      C
      30      219   IF(NWIND.EQ.0)GO TO 290
      30            IF(IP.EQ.IWIND(1))GO TO 286
      30            GO TO 290
      30      286   II=0
      30      C
      30      C     COMPUTE MEAN WIND SPEED(STS(IP,1)) AND DIRECTION
      30      C     (STS(IP+1,1) FROM MEAN COMPONENT VALUES
      30      C
      30            IW=IWIND(1)
      30            WEST=MEAN(IW,ID)
      30            SOUTH=MEAN(IW+1,ID)
      30            STS(IP,1)=DSQRT(WEST**2+SOUTH**2)
      30            IF(ABS(SOUTH).LT.1.D-3.AND.ABS(WEST).LT.1.D-3)STS(IP+1,1)=0.
      30            IF(ABS(SOUTH).LT.1.D-3.AND.ABS(WEST).LT.1.D-3)GO TO 292
      30            IF(ABS(SOUTH).LT.1.D-3.AND.WEST.GE.1.D-3)STS(IF+1,1)=270.
      30            IF(ABS(SOUTH).LT.1.D-3.AND.WEST.LE.-1.D-3)STS(IP+1,1)=90.
      30            IF(ABS(SOUTH).LT.1.D-3)GO TO 292
      30            IF(ABS(WEST).LT.1.D-3.AND.SOUTH.GE.1.D-3)STS(IP+1,1)=180.
      30            IF(ABS(WEST).LT.1.D-3.AND.SOUTH.LE.-1.D-3)STS(IP+1,1)=360.
      30            IF(ABS(WEST).LT.1.D-3)GO TO 292
      30            IF(SOUTH.GE.1.D-3.AND.WEST.LE.-1.D-3)STS(IP+1,1)=DATAN(SOUTH/
      30           *(-WEST))/DTRAD+90.
      30            IF(SOUTH.LE.-1.D-3.AND.WEST.GE.1.D-3)STS(IP+1,1)=DATAN((-SOUTH
      30           *)/WEST)/DTRAD+270.
      30            IF(SOUTH.GE.1.D-3.AND.WEST.GE.1.D-3)STS(IP+1,1)=270.-DATAN
      30           *(SOUTH/WEST)/DTRAD
      30            IF(SOUTH.LE.-1.D-3.AND.WEST.LE.-1.D-3)STS(IP+1,1)=90.-DATAN
      30           *(SOUTH/WEST)/DTRAD
      30            IF(NORMAP)GO TO 291
      30      C
      30      C     COMPUTE VECTOR STANDARD DEVIATION FOR WIND
      30      C
      30      C
      30      C     VSD=VECTOR STANDARD DEVIATION OF WINDS IN CLUSTER
      30      C
      34      292   WEST=SBACK(IW,ID)
      34            SOUTH=SBACK(IW+1,ID)
      30            VSD=DSQRT(WEST**2+SOUTH**2)
      30            STS(IP,2)=VSD
      30            STS(IP+1,2)=999999
```

```
30                STS(IP,3)=STS(IP,1)+2.*STS(IP,2)
30                STS(IP+1,3)=9999999
30                STS(IP,4)=STS(IP,1)-2.*STS(IP,2)
30                STS(IP+1,4)=9999999
30        290     CONTINUE
30        291     LTYP=ID-1
30                PRINT 311,LTYP,LAMBDA(ID)
30                PRINT 310,(I,I=1,NVBLES)
30                IQ=1
30                IF(IUNITS.EQ.1)CALL TABLE(NVBLES,IUNITS,IQ,IW)
30                PRINT 312
30                PRINT 313,(STS(I,1),I=1,NVBLES)
30                IF(NORMAP)GO TO 260
30                PRINT 314
30                PRINT 313,(STS(I,2),I=1,NVBLES)
30                PRINT 316
30                PRINT 313,(STS(I,3),I=1,NVBLES)
30                PRINT 318
30                PRINT 313,(STS(I,4),I=1,NVBLES)
30        316     FORMAT(/53X,28HMEAN + 2 STANDARD DEVIATIONS/)
30        318     FORMAT(/53X,28HMEAN - 2 STANDARD DEVIATIONS/)
30        260     CONTINUE
30        310     FORMAT(/I9,11I11)
30        311     FORMAT(///41X,'CHARACTERISTICS OF TYPE ',I2,1X,'IN BACKTRANSFORMED
30               *VALUES'//41X,'THE PROPORTION OF THE POPULATION FROM THIS TYPE='
30               *,F5.3/)
30        312     FORMAT(/64X,5HMEANS/)
30        313     FORMAT(12F11.4)
30        314     FORMAT(/57X,19HSTANDAPD DEVIATIONS/)
30        C
30        C       FOR A CHI SQUARE TEST FOR NORMALITY AND A HISTOGRAM (BOTH WITH
30        C       EQUAL PROBABILITY INTERVALS) OF THE DATA FOR EACH TRANSFORMED
30        C       VARIATE, CALL THF IMSL SUBROUTINES, GFIT AND USHIST
30        C
30        39      IF(IHIS.NE.1)GO TO 161
30                DO 160 I=2,NTYPES
30                DO 160 IP=1,NVBLES
30                YBAR=0.
30                II=0
30                DO 170 J=1,NSAMPL
30                IF(IGRP(J).NE.I)GO TO 170
30                II=II+1
30                Y(II)=SNGL(X(IP,J))
30                YBAR=YBAR+Y(II)
30        170     CONTINUE
30        C
30                IF(II.LT.20)GO TO 160
30                YBAR=YBAR/FLOAT(II)
30                SIGMA=0.
30                DO 175 J=1,II
30        175     SIGMA=SIGMA+(Y(J)-YBAR)**2
30                SIGMA=SQRT(SIGMA/FLOAT(II-1.))
30                IH=I-1
30                PRINT 105,IH,IP
30        105     FORMAT(///T4,'THE FOLLOWTNG CHI SQUARE TEST AND EQUAL PROBABILITY
30               * HISTOGRAM ARE FOR TYPE NUMBER',I3,' AND VARIABLE NUMBER ',I2)
30                IDF=2
```

```
 284.       30            NUM=II/5
 285.       30            IF(NUM.GT.12)NUM=12
 286.       30            CALL GFIT(NORM,NUM,Y,IT,CELLS,COMP,CS,IDF,Q,IER)
 287.       30            WRITE(6,89)(CELLS(J),J=1,NUM)
 288.       30      89    FORMAT(T2,'CELL NUMBER     1     2     3     4     5     6     7     8',
 289.       30           *'     9    10    11    12'/T2,'FREQUENCY    ',12F5.0)
 290.       30            WRITE(6,88)CS,Q,IDF,SIGMA
 291.       30      88    FORMAT(/T2,'CHI-SQUARE = ',F8.4,'     PROB. OF EXCEEDING CHI-SQUARE
 292.       30           * IF NULL HYPOTHESIS IS TRUE =',F8.4,'     DF = ',I4/T2,
 293.       30           *'STANDARD DEVIATION OF THIS VARIABLE = ',F8.4)
 294.       30            ISPACE=4
 295.       30            CALL USHIST(CELLS,NUM,ISPACE,WORK1,WORK2,IWORK,IER)
 296.       30      160   CONTINUE
 297.       30      161   LIKLY=AL                                                          00001150
 298.       30            IF(NORMAP) RETURN                                                 00001160
 299.       30            DO 34  L=2,NTYPES                                                 00001170
 300.       30            DO 34 I=1,NSEMI                                                   00001180
 301.       30      34    COVIN(I,L)=MU(I,L)                                                00001190
 302.       30        99  RETURN                                                            00001200
 303.       30      C                                                                       00001210
 304.       30      14    FORMAT (//25X,22HDISCRIMINANT FUNCTIONS/(16X,10I10))              00001220
 305.       30      15    FORMAT (9H VARIABLEI4,7X,10F11.4/(20X,10F11.4))                   00001230
 306.       30      16    FORMAT (/20H EIGENVALUES OF B/W=10F11.4)                          00001240
 307.       30      17    FORMAT(  9X,32HPROBABILITIES OF TYPE MEMBERSHIP)                  00001250
 308.       30      171   FORMAT(1H+,73X,19HDISCRIMINANT SCORES)                            00001260
 309.       30      172   FORMAT( 1H  )                                                     00001270
 310.       30      173   FORMAT(70X,8I6)                                                   00001280
 311.       30      18    FORMAT(1H+,I10,9I6/)                                              00001290
 312.       30      19    FORMAT (1H ,I7,10F6.3)                                            00001300
 313.       30      20    FORMAT (I6,I4,10F7.2)                                             00001310
 314.       30      21    FORMAT(1H+,71X,8F6.2)                                             00001320
 315.       30      22    FORMAT (8X,10F6.3)                                                00001330
 316.       30            END                                                               00001340

END ELT.  ERRORS: NONE. TIME:  2.668 SEC. IMAGE COUNT: 316

@FIN
```

```
9 161    LIKLY=AL                                                       00001150
9        IF(NORMAP) RETURN                                              00001160
9        DO 34  L=2,NTYPES                                              00001170
9        DO 34 I=1,NSEMI                                                00001180
9 34     COVIN(I,L)=MU(I,L)                                             00001190
9   99   RETURN                                                         00001200
9 C                                                                     00001210
9 14     FORMAT (//25X,22HDISCRIMINANT FUNCTIONS/(16X,10I10))           00001220
9 15     FORMAT (9H VARIABLEI4,7X,10F11.4/(20X,10F11.4))                00001230
9 16     FORMAT (/20H EIGENVALUES OF B/W=10F11.4)                       00001240
9 17     FORMAT(  9X,32HPROBABILITIES OF TYPE MEMBERSHIP)               00001250
9 171    FORMAT(1H+,73X,19HDISCRIMINANT SCORES)                         00001260
9 172    FORMAT( 1H  )                                                  00001270
9 173    FORMAT(70X,8I6)                                                00001280
9 18     FORMAT(1H+,I10,9I6/)                                           00001290
9 19     FORMAT (1H ,I7,10F6.3)                                         00001300
9 20     FORMAT (I6,I4,10F7.2)                                          00001310
9 21     FORMAT(1H+,71X,8F6.2)                                          00001320
9 22     FORMAT (6X,10F6.3)                                             00001330
9        END                                                            00001340

END ELT.
```

```
@ELT,L N.CHISQP/CLUSTER
EL1007 SL73R1 02/25/80 16:08:55 (1,)
000001       000            FUNCTION CHISQP (CHISQ,NDFREE)                                      00001350
000002       000            IMPLICIT REAL*8 (A-H,O-Z)                                           00001360
000003       000            A=CHISQ*0.5                                                         00001380
000004       000            T=DEXP(-A)                                                          00001390
000005       000            NN=(NDFREE/2)*2                                                     00001400
000006       000            IF (NN-NDFREE) 2,1,2                                                00001410
000007       000      1     P=T                                                                 00001420
000008       000            T=A*T                                                               00001430
000009       000            KW=4                                                                00001440
000010       000            AK=2.0                                                              00001450
000011       000            GO TO 3                                                             00001460
000012       000      2     C=DSQRT(CHISQ)                                                      00001470
000013       000            P=2.0*AREA(C)                                                       00001480
000014       000            T=0.7978846*C*T                                                     00001490
000015       000            AK=1.5                                                              00001500
000016       000            KW=3                                                                00001510
000017       000      3     IF (NDFREE-2) 6,6,4                                                 00001520
000018       000      4     DO 5 I=KW,NDFREE,2                                                  00001530
000019       000            P=P+T                                                               00001540
000020       000            T=(T*A)/AK                                                          00001550
000021       000      5     AK=1.0+AK                                                           00001560
000022       000      6     CHISQP=P                                                            00001570
000023       000            RETURN                                                              00001580
000024 .     000            END                                                                 00001590

END ELT.
```

```
@FLT,L N.DENSIT/CLUSTER
EL1007 SL73R1 02/25/80 16:08:56 (21,)
000001         7       SUBROUTINE DENSIT (MEAN,COVIN,X,R,S,B,V,G,K)                00000010
000002         7       IMPLICIT REAL*8 (A-H,O-Z)                                   00000020
000003         7       REAL*8 LAMBDA,MEAN,MU,LIKLY
000004         7       LOGICAL NORMAP                                              00000050
000005         7       COMMON/A/LIKLY,XLIMIT,NVBLES,NSAMPL,NTYPES,NSEMI,NPARAM,NORMAP,KX 00000060
000006         7       COMMON/B/CONVRG,CRITRN,RATIO,HTEST,OLDLIK,EN,BIGEST,THRESH,GUARD, 00000070
000007         7      1ENMIN,TITLE(18),FMT( 8),    MAXTYP,MAXAV,NCLUST,KSCRAT,INP,KPUN,KV,00000080
000008         7      2 LOT,LACK,LUCK,IPRINT,LTIME,MAXT,ITER,ITERX,LTYE,ITMC,NRANK,IVALUE00000090
000009         7      3,METHOD,NTYP(20)
000010         7       DIMENSION MEAN(1),COVIN(1),X(1),R(1),                       00000110
000011         7      1 S(1),B(1),G(1),V(1)                                        00000120
000012         7       IF(K .GT. 1) GO TO 50                                       00000130
000013         7       XLIMIT=40.0D0                                               00000140
000014         7       N=NVBLES-1                                                  00000150
000015         7       ASSIGN 3 TO LONE                                            00000160
000016         7       IF (N.EQ.0) ASSIGN 6 TO LONE                                00000170
000017         7       IK=1                                                        00000180
000018         7 50    EPH=0.0D0                                                   00000190
000019         7       DO 1 I=1,NVBLES                                             00000200
000020         7       R(I)=X(IK)                                                  00000210
000021         7 1     IK=IK+1                                                     00000220
000022         7       IL=NVBLES                                                   00000230
000023         7       IF(NORMAP) GO TO 40                                         00000240
000024         7       IJ=NSEMI                                                    00000250
000025         7       DO 7 L=2,NTYPES                                             00000260
000026         7       DO 2 I=1,NVBLES                                             00000270
000027         7       IL=IL+1                                                     00000280
000028         7 2     S(I)=R(I)-MEAN(IL)                                          00000290
000029         7       DSQRD=  B(L)                                                00000300
000030         7       GO TO LONE, (3,6)                                           00000310
000031         7 3     DO 5 I=1,N                                                  00000320
000032         7       IJ=IJ+1                                                     00000330
000033         7       CUTE=S(I)*COVIN(IJ)                                         00000340
000034         7       DO 4 J=I,N                                                  00000350
000035         7       IJ=IJ+1                                                     00000360
000036         7 4     CUTE=CUTE+COVIN(IJ)*S(J+1)                                  00000370
000037         7 5     DSQRD=DSQRD-CUTE*S(I)                                       00000380
000038         7 6     IJ=IJ+1                                                     00000390
000039         7 7     G(L)=DSQRD-COVIN(IJ)*S(NVBLES)**2                           00000400
000040         7       GO TO 43                                                    00000410
000041         7 40    DO 42 L=2,NTYPES                                            00000420
000042         7       DSQRD= B(L)                                                 00000430
000043         7       DO 41 I=1,NVBLES                                            00000440
000044         7       IL=IL+1                                                     00000450
000045         7 41    DSQRD=DSQRD+V(IL)*R(I)                                      00000460
000046         7 42    G(L)=DSQRD                                                  00000470
000047         7 43    BIGA=G(2)                                                   00000480
000048         7       LOT=2                                                       00000490
000049         7       BIGB=BIGA                                                   00000500
```

```
000050        7         IF (NTYPES.EQ.2) GO TO 13                                     00000510
000051        7         DO 9 L=3,NTYPES                                               00000520
000052        7         IF (G(L).LE.BIGB) GO TO 9                                     00000530
000053        7         IF (G(L).LE.BIGA) GO TO 8                                     00000540
000054        7         BIGB=BIGA                                                     00000550
000055        7         BIGA=G(L)                                                     00000560
000056        7         LOT=L                                                         00000570
000057        7         GO TO 9                                                       00000580
000058        7 8       BIGB=G(L)                                                     00000590
000059        7 9       CONTINUE                                                      00000600
000060        7         IF ((BIGA-BIGB).GT.XLIMIT) GO TO 13                           00000610
000061        7         G(LOT)=1.0D0                                                  00000620
000062        7         DO 10 L=2,NTYPES                                              00000630
000063        7         IF (L.EQ.LOT) GO TO 10                                        00000640
000064        7         DSQRD=G(L)-BIGA                                               00000650
000065        7         G(L)=0.0D0                                                    00000660
000066        7         IF (DSQRD.GT.-4D1    ) G(L)=DEXP(DSQRD)                       00000670
000067        7 10      EPH=EPH+G(L)                                                  00000680
000068        7         LIKLY=LIKLY+DLOG(EPH)+BIGA                                    00000690
000069        7         DO 11 L=2,NTYPES                                              00000700
000070        7 11      G(L)=G(L)/EPH                                                 00000710
000071        7         RETURN                                                        00000720
000072        7 C       CONSIDER THE PROBABILITY OF MEMBERSHIP TO BE ONE FOR NEAREST TYPE 00000730
000073        7 13      LIKLY=LIKLY+BIGA                                              00000740
000074        7         DO 14 L=2,NTYPES                                              00000750
000075        7 14      G(L)=0.0D0                                                    00000760
000076        7         G(LOT)=1.0D0                                                  00000770
000077        7         RETURN                                                        00000780
000078        7         END                                                           00000790

END ELT.
```

```
@ELT,L N.DUMPER/CLUSTER
ELT007 SL73R1 02/25/80 16:08:58 (12,)
000001          8        SUBROUTINE DUMPER (LAMBDA,MEAN,COVIN,R,S)                          00000010
000002          8        IMPLICIT REAL*8 (A-H,O-Z)                                          00000020
000003          8        REAL*8 DSQRT                                                       00000030
000004          8        REAL*8 LAMBDA,MEAN,MU,LIKLY
000005          8        LOGICAL NORMAP                                                     00000060
000006          8        COMMON/A/LIKLY,XLIMIT,NVBLES,NSAMPL,NTYPES,NSEMI,NPARAM,NORMAP,KX 00000070
000007          8        COMMON/B/CONVRG,CRITRN,RATIO,HTEST,OLDLIK,EN,BIGEST,THRESH,GUARD, 00000080
000008          8       1ENMIN,TITLE(18),FMT( 8),    MAXTYP,MAXAV,NCLUST,KSCRAT,INP,KPUN,KV,00000090
000009          8       2 LOT,LACK,LUCK,IPRINT,LTIME,MAXT,ITER,ITERX,LTYE,ITMC,NRANK,IVALUE00000100
000010          2       3,METHOD,NTYP(20)
000011          8        DIMENSION MEAN(NVBLES,NTYPES),COVIN(NSEMI,NTYPES),LAMBDA(1),       00000120
000012          8       1 R(1),S(1)                                                         00000130
000013          8        N=2                                                                00000140
000014          8 C      THIS ROUTINE PUNCHES PARAMETERS ON CARDS IN FORMAT READABLE BY THE00000150
000015          0 C      SUBROUTINE REVALU.                                                 00000010
000016          8        DO 5 L=2,NTYPES                                                    00000020
000017          8        IF(.NOT. NORMAP) N=L                                               00000030
000018          8        LTYP=L-1                                                           00000040
000019          8        WRITE(KPUN,6) LAMBDA(L),LTYP                                       00000050
000020          8        WRITE(KPUN,7) (MEAN(I,L),I=1,NVBLES)                               00000060
000021          8        LQQ=1                                                              00000070
000022          8        DO 1 I=1,NVBLES                                                    00000080
000023          8        S(I)=DSQRT(COVIN(LQQ,N))                                           00000090
000024          8 1      LQQ=LQQ+NVBLES-I+1                                                 00000100
000025          8        WRITE(KPUN,7) (S(I),I=1,NVBLES)                                    00000110
000026          8        DO 5 I=1,NVBLES                                                    00000120
000027          8        KJ=I-1                                                             00000130
000028          8        IJ=I                                                               00000140
000029          8        IF (KJ.EQ.0) GO TO 3                                               00000150
000030          8        DO 2 J=1,KJ                                                        00000160
000031          8        R(J)=COVIN(IJ,N)/(S(I)*S(J))                                       00000170
000032          8 2      IJ=IJ+NVBLES-J                                                     00000180
000033          8 3      DO 4 J=I,NVBLES                                                    00000190
000034          8        R(J)=COVIN(IJ,N)/(S(I)*S(J))                                       00000200
000035          8 4      IJ=IJ+1                                                            00000210
000036          8 5      WRITE(KPUN,7) (R(J),J=1,NVBLES)                                    00000220
000037          8        RETURN                                                             00000230
000038          8 C                                                                         00000240
000039          8 6      FORMAT (F8.4,5X,4HTYPEI3)                                          00000250
000040          8 7      FORMAT (10F8.4)                                                    00000260
000041          8        END                                                                00000270

END ELT.
```

```
@ELT,L N.EIGEN/CLUSTER
ELT007 SL73R1 02/25/80 16:08:59 (18,)
000001        014           SUBROUTINE EIGEN(A,R,N,MV)                                     00000010
000002        014           REAL*8 DSQRT                                                   00000020
000003        014    C                                                                     00000030
000004        014    C     ...............................................................00000040
000005        014    C                                                                     00000050
000006        014    C        SUBROUTINE EIGEN                                             00000060
000007        014    C                                                                     00000070
000008        014    C        PURPOSE                                                      00000080
000009        014    C           COMPUTE EIGENVALUES AND EIGENVECTORS OF A REAL SYMMETRIC  00000090
000010        014    C           MATRIX                                                    00000100
000011        014    C                                                                     00000110
000012        014    C        USAGE                                                        00000120
000013        014    C           CALL EIGEN(A,R,N,MV)                                      00000130
000014        014    C                                                                     00000140
000015        014    C        DESCRIPTION OF PARAMETERS                                    00000150
000016        014    C           A - ORIGINAL MATRIX (SYMMETRIC), DESTROYED IN COMPUTATION. 00000160
000017        014    C               RESULTANT EIGENVALUES ARE DEVELOPED IN DIAGONAL OF    00000170
000018        014    C               MATRIX A IN DESCENDING ORDER.                         00000180
000019        014    C           R - RESULTANT MATRIX OF EIGENVECTORS (STORED COLUMNWISE,  00000190
000020        014    C               IN SAME SEQUENCE AS EIGENVALUES)                      00000200
000021        014    C           N - ORDER OF MATRICES A AND R                             00000210
000022        014    C           MV- INPUT CODE                                            00000220
000023        014    C                   0   COMPUTE EIGENVALUES AND EIGENVECTORS          00000230
000024        014    C                   1   COMPUTE EIGENVALUES ONLY (R NEED NOT BE       00000240
000025        014    C                       DIMENSIONED BUT MUST STILL APPEAR IN CALLING  00000250
000026        014    C                       SEQUENCE)                                     00000260
000027        014    C                                                                     00000270
000028        014    C        REMARKS                                                      00000280
000029        014    C           ORIGINAL MATRIX A MUST BE REAL SYMMETRIC (STORAGE MODE=1) 00000290
000030        014    C           MATRIX A CANNOT BE IN THE SAME LOCATION AS MATRIX R       00000300
000031        014    C                                                                     00000310
000032        014    C        SUBROUTINES AND FUNCTION SUBPROGRAMS REQUIRED                00000320
000033        014    C           NONE                                                      00000330
000034        014    C                                                                     00000340
000035        014    C        METHOD                                                       00000350
000036        014    C           DIAGONALIZATION METHOD ORIGINATED BY JACOBI AND ADAPTED   00000360
000037        014    C           BY VON NEUMANN FOR LARGE COMPUTERS AS FOUND IN 'MATHEMATICAL00000370
000038        014    C           METHODS FOR DIGITAL COMPUTERS', EDITED BY A. RALSTON AND  00000380
000039        014    C           H.S. WILF, JOHN WILEY AND SONS, NEW YORK, 1962, CHAPTER 7 00000390
000040        014    C                                                                     00000400
000041        014    C     ...............................................................00000410
000042        014    C                                                                     00000420
000043        014    C                                                                     00000440
000044        014    C                                                                     00000450
000045        014    C        ............................................................00000450
000046        014    C                                                                     00000460
000047        014    C        IF A DOUBLE PRECISION VERSION OF THIS ROUTINE IS DESIRED, THE 00000470
000048        014    C        C IN COLUMN 1 SHOULD BE REMOVED FROM THE DOUBLE PRECISION    00000480
000049        014    C        STATEMENT WHICH FOLLOWS.                                     00000490
                                                                                           00000500
```

```
000050    014         DOUBLE PRECISION A,R,ANORM,ANRMX,THR,X,Y,SINX,SINX2,COSX,      00000510
000051    014        1                COSX2,SINCS,RANGE                              00000520
000052    016         COMMON/D/IHIER,MERGE,IGEN,IZQ,IHIS
000053    014         DIMENSION A(1),R(1),KP(20)
000054    014   C                                                                    00000530
000055    014   C        THE C MUST ALSO BE REMOVED FROM DOUBLE PRECISION STATEMENTS    00000540
000056    014   C        APPEARING IN OTHER ROUTINES USED IN CONJUNCTION WITH THIS      00000550
000057    014   C        ROUTINE.                                                      00000560
000058    014   C                                                                    00000570
000059    014   C        THE DOUBLE PRECISION VERSION OF THIS SUBROUTINE MUST ALSO      00000580
000060    014   C        CONTAIN DOUBLE PRECISION FORTRAN FUNCTIONS.  SQRT IN STATEMENTS00000590
000061    014   C        40, 68, 75, AND 78 MUST BE CHANGED TO DSQRT.  ABS IN STATEMENT 00000600
000062    014   C        62 MUST BE CHANGED TO DABS. THE CONSTANT IN STATEMENT 5 SHOULD 00000610
000063    014   C        BE CHANGED TO 1.0D-12.                                        00000620
000064    014   C                                                                    00000630
000065    014   C        ...............................................................00000640
000066    014   C                                                                    00000650
000067    014   C        GENERATE IDENTITY MATRIX                                      00000660
000068    014   C                                                                    00000670
000069    014       5 RANGE=1.0D-12                                                  00000680
000070    014         IF(MV-1) 10,25,10                                              00000690
000071    014      10 IQ=-N                                                          00000700
000072    014         DO 20 J=1,N                                                    00000710
000073    014         IQ=IQ+N                                                        00000720
000074    014         DO 20 I=1,N                                                    00000730
000075    014         IJ=IQ+I                                                        00000740
000076    014         R(IJ)=0.0                                                      00000750
000077    014         IF(I-J) 20,15,20                                               00000760
000078    014      15 R(IJ)=1.0                                                      00000770
000079    014      20 CONTINUE                                                       00000780
000080    014   C                                                                    00000790
000081    014   C        COMPUTE INITIAL AND FINAL NORMS (ANORM AND ANORMX)            00000800
000082    014   C                                                                    00000810
000083    014      25 ANORM=0.0                                                      00000820
000084    014         DO 35 I=1,N                                                    00000830
000085    014         DO 35 J=I,N                                                    00000840
000086    014         IF(I-J) 30,35,30                                               00000850
000087    014      30 IA=I+(J*J-J)/2                                                 00000860
000088    014         ANORM=ANORM+A(IA)*A(IA)                                        00000870
000089    014      35 CONTINUE                                                       00000880
000090    014         IF(ANORM) 165,165,40                                           00000890
000091    014      40 ANORM=1.414D0*DSQRT(ANORM)                                     00000900
000092    014         ANRMX=ANORM*RANGE/DFLOAT(N)                                    00000910
000093    014   C                                                                    00000920
000094    014   C        INITIALIZE INDICATORS AND COMPUTE THRESHOLD, THR              00000930
000095    014   C                                                                    00000940
000096    014         IND=0                                                          00000950
000097    014         THR=ANORM                                                      00000960
000098    014      45 THR=THR/DFLOAT(N)                                              00000970
000099    014      50 L=1                                                            00000980
```

```
014         55 M=L+1
014      C
014      C        COMPUTE SIN AND COS
014      C
014         60 MQ=(M*M-M)/2
014            LQ=(L*L-L)/2
014            LM=L+MQ
014         62 IF(DABS(A(LM))-THR) 130,65,65
014         65 IND=1
014            LL=L+LQ
014            MM=M+MQ
014            X=0.5*(A(LL)-A(MM))
014         68 Y=-A(LM)/DSQRT(A(LM)*A(LM)+X*X)
014            IF(X) 70,75,75
014         70 Y=-Y
014         75 SINX=Y/DSQRT(2.0*(1.0+(DSQRT(1.0-Y*Y))))
014            SINX2=SINX*SINX
014         78 COSX=DSQRT(1.0-SINX2)
014            COSX2=COSX*COSX
014            SINCS =SINX*COSX
014      C
014      C        ROTATE L AND M COLUMNS
014      C
014            ILQ=N*(L-1)
014            IMQ=N*(M-1)
014            DO 125 I=1,N
014            IQ=(I*I-I)/2
014            IF(I-L) 80,115,80
014         80 IF(I-M) 85,115,90
014         85 IM=I+MQ
014            GO TO 95
014         90 IM=M+IQ
014         95 IF(I-L) 100,105,105
014        100 IL=I+LQ
014            GO TO 110
014        105 IL=L+IQ
014        110 X=A(IL)*COSX-A(IM)*SINX
014            A(IM)=A(IL)*SINX+A(IM)*COSX
014            A(IL)=X
014        115 IF(MV-1) 120,125,120
014        120 ILR=ILQ+I
014            IMR=IMQ+I
014            X=R(ILR)*COSX-R(IMR)*SINX
014            R(IMR)=R(ILR)*SINX+R(IMR)*COSX
014            R(ILR)=X
014        125 CONTINUE
014            X=2.0*A(LM)*SINCS
014            Y=A(LL)*COSX2+A(MM)*SINX2-X
014            X=A(LL)*SINX2+A(MM)*COSX2+X
014            A(LM)=(A(LL)-A(MM))*SINCS+A(LM)*(COSX2-SINX2)
```

```
000150    014            A(LL)=Y                                                  00001490
000151    014            A(MM)=X                                                  00001500
000152    014   C                                                                 00001510
000153    014   C            TESTS FOR COMPLETION                                 00001520
000154    014   C                                                                 00001530
000155    014   C            TEST FOR M = LAST COLUMN                             00001540
000156    014   C                                                                 00001550
000157    014       130 IF(M-N) 135,140,135                                       00001560
000158    014       135 M=M+1                                                     00001570
000159    014            GO TO 60                                                 00001580
000160    014   C                                                                 00001590
000161    014   C            TEST FOR L = SECOND FROM LAST COLUMN                 00001600
000162    014   C                                                                 00001610
000163    014       140 IF(L-(N-1)) 145,150,145                                   00001620
000164    014       145 L=L+1                                                     00001630
000165    014            GO TO 55                                                 00001640
000166    014       150 IF(IND-1) 160,155,160                                     00001650
000167    014       155 IND=0                                                     00001660
000168    014            GO TO 50                                                 00001670
000169    014   C                                                                 00001680
000170    014   C           COMPARE THRESHOLD WITH FINAL NORM                     00001690
000171    014   C                                                                 00001700
000172    014       160 IF(THR-ANRMX) 165,165,45                                  00001710
000173    014   C                                                                 00001720
000174    014   C           SORT EIGENVALUES AND EIGENVECTORS                     00001730
000175    014   C                                                                 00001740
000176    014       165 IQ=-N                                                     00001750
000177    014            DO 185 I=1,N                                             00001760
000178    014            IQ=IQ+N                                                  00001770
000179    014            LL=I+(I*I-I)/2                                           00001780
000180    014            JQ=N*(I-2)                                               00001790
000181    014            DO 185 J=I,N                                             00001800
000182    014            JQ=JQ+N                                                  00001810
000183    014            MM=J+(J*J-J)/2                                           00001820
000184    014            IF(A(LL)-A(MM)) 170,185,185                              00001830
000185    014       170 X=A(LL)                                                   00001840
000186    014            A(LL)=A(MM)                                              00001850
000187    014            A(MM)=X                                                  00001860
000188    014            IF(MV-1) 175,185,175                                     00001870
000189    014       175 DO 180 K=1,N                                              00001880
000190    014            ILR=IQ+K                                                 00001890
000191    014            IMR=JQ+K                                                 00001900
000192    014            X=R(ILR)                                                 00001910
000193    014            R(ILR)=R(IMR)                                            00001920
000194    014       180 R(IMR)=X                                                  00001930
000195    014       185 CONTINUE                                                  00001940
000196    015            IF(IGEN.NE.1)GO TO 199
000197    015            WRITE(6,195)
000198    015       195 FORMAT(1H1,T8,'EIGENVALUES',T57,'EIGENVECTORS BY ROWS')
000199    015            IP=0
```

```
015          DO 190 I=1,N
015          DO 192 II=1,I
015      192 IP=IP+1
015          WRITE(6,200)A(IP)
018      200 FORMAT(/T9,F8.3)
015          DO 191 J=1,N
015      191 KP(J)=(I-1)*4+J
015      190 WRITE(6,205)(R(KP(K)),K=1,N)
017      205 FORMAT(1H+,T20,12F8.3)
015      199 RETURN                                          00001950
014          END                                             00001960

END ELT.
```

```
@ELT,L N.HISTOG/CLUSTER
EL1007 SL73R1 02/25/80 16:09:01 (31,)
000001        7        SUBROUTINE HISTOG(MEAN,AV,DENSE,X,R,S,G,B,MU,COVIN,T,MEMBER,KPATH)00000010
000002        7        IMPLICIT REAL*8 (A-H,O-Z)                                         00000060
000003        7 C      THE CLUSTERS WHICH ARE FORMED ARE USED LATER IN THE PROGRAM FOR   00000040
000004        7 C      INITIAL ESTIMATES OF THE PARAMETERS.                              00000050
000005        7        REAL*8 LAMBDA,MEAN,MU,LIKLY                                       00000070
000006        7        LOGICAL NORMAP,KMAX                                               00000090
000007        7        INTEGER B,G                                                       00000100
000008        7        COMMON/A/LIKLY,XLIMIT,NVBLES,NSAMPL,NTYPES,NSEMI,NPARAM,NORMAP,KX 00000110
000009        7        COMMON/B/CONVRG,CRITRN,RATIO,HTEST,OLDLIK,EN,BIGEST,THRESH,GUARD, 00000120
000010        7       1ENMIN,TITLE(18),FMT( 8),    MAXTYP,MAXAV,NCLUST,KSCRAT,INP,KPUN,KV,00000130
000011        7       2 LOT,LACK,LUCK,IPRINT,LTIME,MAXT,ITER,ITERX,LTYE,ITMC,NRANK,IVALUE00000140
000012        1       3,METHOD,NTYP(20)
000013        9        COMMON/D/IHIER,MERGE,IGEN,IZQ,IHIS
000014        7        DIMENSION MEAN(NVBLES,MAXTYP),X(NVBLES,NSAMPL),AV(NVBLES,MAXTYP), 00000160
000015        7       1 DENSE(1),R(1),G(1),MU(1),COVIN(NSEMI,MAXTYP),                    00000170
000016        7       2S(1),MEMBER(1),B(1),T(NVBLES,NVBLES),KPATH(1)                     00000180
000017        7        DO 2 I=1,NSEMI                                                    00000190
000018        7 2      MU(I)=COVIN(I,1)                                                  00000200
000019        7        KV=2                                                              00000210
000020        7        KC=3                                                              00000220
000021        7        NGRP=0                                                            00000230
000022        7     50 NGRP=NGRP+1                                                       00000240
000023        7        REWIND KV                                                         00000250
000024        7        REWIND KC                                                         00000260
000025        7        CALL UPLOW (MU,T,NVBLES)                                          00000270
000026        7        CALL EIGEN(MU,T,NVBLES,0)                                         00000280
000027        7        WRITE (KV) X                                                      00000290
000028        7        II=0                                                              00000300
000029        7        DO 4 I=1,NVBLES                                                   00000310
000030        7        II=II+1                                                           00000320
000031        7        IF (MU(II).LE.0.) GO TO 300                                       00000330
000032        7   302  RA=1.0D0/DSQRT(MU(II))                                            00000340
000033        7        DO 3 J=1,NVBLES                                                   00000350
000034        7 3      T(J,I)=T(J,I)*RA                                                  00000360
000035        7 4      CONTINUE                                                          00000370
000036        7        DO 21 K=1,NSAMPL                                                  00000390
000037        7        DO 20 I=1,NVBLES                                                  00000400
000038        7        R(I)=0.0                                                          00000410
000039        7        DO 20 J=1,NVBLES                                                  00000420
000040        7 20     R(I)=R(I)+T(J,I)*(X(J,K)-MEAN(J,1))                               00000430
000041        7        WRITE (KC) (R(L),L=1,NVBLES)                                      00000440
000042        7     21 CONTINUE                                                          00000450
000043        7        CALL KMEAN(AV,DENSE,X,S,MEMBER,B,G,KPATH,MAXAV,NSAMPL,NVBLES,KC)  00000460
000044        7        REWIND KV                                                         00000470
000045        7        READ  (KV) X                                                      00000480
000046        7        DO 24 L=1,MAXAV                                                   00000490
000047        7        DO 24 I=1,NVBLES                                                  00000500
000048        7 24     AV(I,L)=0.0                                                       00000510
```

```
000049        7          L=0
000050        7          DO 28 K=1,NSAMPL                                         00000520
000051        7          KI=KPATH(K)                                              00000530
000052        7          IF(KI.GT.0) GO TO 26                                     00000540
000053        7          KI=-KI                                                   00000550
000054        7          L=L+1                                                    00000560
000055        7 26       DO 28 I=1,NVBLES                                         00000570
000056        7 28       AV(I,L)=AV(I,L)+X(I,KI)                                  00000580
000057        7          IF(NGRP .GT. 2) GO TO 59                                 00000590
000058        7          LTYPES=MINO(10,MAXAV)                                    00000600
000059        7          D=0.0                                                    00000610
000060        7          IJ=0                                                     00000620
000061        7          DO 51 I=1,NVBLES                                         00000630
000062        7          R(I)=0.0                                                 00000640
000063        7          DO 51 J=I,NVBLES                                         00000650
000064        7          IJ=IJ+1                                                  00000660
000065        7       51 MU(IJ)=COVIN(IJ,1)-MEAN(I,1)*MEAN(J,1)                   00000670
000066        7          KMAX=.FALSE.                                             00000680
000067        7          DO 57 K=1,MAXAV                                          00000690
000068        7          IF(MEMBER(K) .GE.LTYPES) GO TO 55                        00000700
000069        7          IF(D.LE. 0.5) GO TO 55                                   00000710
000070        7       58 IJ=0                                                     00000720
000071        7          DO 53 I=1,NVBLES                                         00000730
000072        7          DO 53 J=I,NVBLES                                         00000740
000073        7          IJ=IJ+1                                                  00000750
000074        7       53 MU(IJ)=MU(IJ)-R(I)*R(J)/(D*EN)                           00000760
000075        7          IF(KMAX) GO TO 50                                        00000770
000076        7          D=0.0                                                    00000780
000077        7          DO 54 I=1,NVBLES                                         00000790
000078        7       54 R(I)=0.0                                                 00000800
000079        7       55 D=D+DENSE(K)                                             00000810
000080        7          DO 56 I=1,NVBLES                                         00000820
000081        7       56 R(I)=R(I)+AV(I,K)                                        00000830
000082        7          KMAX=K.EQ.MAXAV                                          00000840
000083        7          IF(KMAX) GO TO 58                                        00000850
000084        7       57 CONTINUE                                                 00000860
000085        7          GO TO 50                                                 00000870
000086        7  300     IK = II-1                                                00000880
000087        7          TEM = MU(IK)/2.                                          00000890
000088        7          PRINT 301, I,II,NVBLES,MU(II),TEM                        00000900
000089        7  301     FORMAT (3I7,F12.5,' NEG EIGENVALUE MADE=',F12.5)         00000910
000090        7          MU(II)= TEM                                              00000920
000091        7          GO TO 302                                                00000930
000092        7       59 DO 40 L=1,MAXAV                                          00000940
000093        7          DO 40 I=1,NVBLES                                         00000950
000094        7       40 AV(I,L)=AV(I,L)/DENSE(L)                                 00000960
000095        7          IF(MERGE.EQ.1)CALL TYTLE                                 00000970
000096        7          IF(MERGE.EQ.1)PRINT 1140                                 00000980
000097        7          LAST=0                                                   00000990
000098        7          DO 440 I=1,MAXAV                                         00001000
                                                                                  00001010
```

```
7        K=LAST+1                                                      00001020
7        L=DENSE(I)                                                    00001030
7        LAST=L+K-1                                                    00001040
7 C      PRINT 1130,I,L,(KPATH(J),J=K,LAST)                            00001050
7 C      PRINT 1150,(AV(J,I),J=1,NVBLES)                               00001060
7 440    CONTINUE
7        DO 35 MTYPES=2,MAXAV                                          00001070
7        LTYPES=MAXAV-MTYPES+2                                         00001080
7        LAST=G(LTYPES)                                                00001090
7        J=LAST                                                        00001100
7        JIND=1                                                        00001110
7        DO 29 I=1,NVBLES                                              00001120
7 29     R(I)=AV(I,J)*DENSE(J)                                         00001130
7        AWT=DENSE(J)                                                  00001140
7 30     J=J+JIND                                                      00001150
7        IF(J.GT.MAXAV) GO TO 32                                       00001160
7        IF(MEMBER(J).LT.LTYPES*JIND) GO TO 32                         00001170
7        DO 31 I=1,NVBLES                                              00001180
7 31     R(I)=R(I)+AV(I,J)*DENSE(J)                                    00001190
7        AWT=AWT+DENSE(J)                                              00001200
7        IF(MEMBER(J).GE.LTYPES) GO TO 30                              00001210
7 32     IF(J.LT.LAST) GO TO 33                                        00001220
7        J=LAST                                                        00001230
7        JIND=-1                                                       00001240
7        GO TO 30                                                      00001250
7 33     DO 34 I=1,NVBLES                                              00001260
7 34     R(I)=R(I)/AWT                                                 00001270
7        IF(MERGE.EQ.1)PRINT 1160,LAST,J                               00001280
7        NJ=AWT                                                        00001290
7        NJ1=B(J)                                                      00001300
7        NJ2=NJ1+NJ-1                                                  00001310
7        IF(MERGE.EQ.1)PRINT 1130,J,NJ,(KPATH(M),M=NJ1,NJ2)
7 35     CONTINUE
7 1130   FORMAT(/30X,'CLUSTER',I4,10X,I4,' POINTS'/50(1X,30I4/))       00001340
7 1140   FORMAT(42X,'KMEANS ANALYSIS'//)                               00001350
7 1150   FORMAT(10F12.4)                                               00001360
7 1160   FORMAT(/20X,'MERGING CLUSTER'I4,' WITH CLUSTER'I4,' RESULTS IN THE00001370
7       1 REVISED' )                                                   00001380
7        RETURN                                                        00001390
7 C                                                                    00001400
7        END                                                           00001410

END ELT.
```

```
@ELT,L N. INFORM/CLUSTER
ELT007 SL73R1 04/07/80 13:49:47 (4,)
000001         000          SUBROUTINE INFORM                                                  00000010
000002         000    C     THIS SUBROUTINF READS THE INPUT FORM CARDS ON WHICH THE USER       00000020
000003         000    C     SPECIFIES THE OPTIONS CONTROLLING THE PROGRAM.                     00000030
000004         000          IMPLICIT REAL*8 (A-H,O-Z)                                          00000040
000005         000          REAL*8 LAMBDA,MEAN,MU,LIKLY.
000006         000          LOGICAL NORMAP                                                     00000070
000007         000          COMMON/A/LIKLY,XLIMIT,NVBLES,NSAMPL,NTYPES,NSEMT,NPARAM,NORMAP,KX 00000080
000008         000          COMMON/B/CONVRG,CRITRN,RATIO,HTEST,OLDLIK,FN,BIGEST,THRESH,GUARD, 00000090
000009         000         1ENMIN,TITLE(18),FMT( 8),    MAXTYP,MAXAV,NCLUST,KSCRAT,TNP,KPUN,KV,00000100
000010         000         2 LOT,LACK,LUCK,IPRINT,LTIME,MAXT,ITER,ITERX,LTYE,ITMC,NRANK,IVALUE00000110
000011         000         3,METHOD,NTYP(20)
000012         000          COMMON/D/IHIER,MERGE,IGEN,IZQ,IHIS
000013         000          COMMON/E/MTIME,NTIME,IOVER
000014         000          COMMON/F/IMAP,IUNITS,IPROF,IBACK,IPOL(20),SCAL(20),POWER,IWIND(2),
000015         000         *NWIND
000016         000          COMMON/G/IMN,ISTD,ICOR
000017         000          NAMELIST/LIST/NVBLES,NSAMPL,INP,NTYP,NOR,ENMIN,HTEST,MAXAV,IPRINT,
000018         000         *LTIME,ITERX,IHIER,MERGE,IGEN,MTIME,NTIME,IHIS,IOVER,IMAP,IBACK,
000019         000         *IUNITS,IPROF,IPOL,POWER,NWIND,IWIND
000020         000          DATA IDIF/'DIFF'/,ISAME/'SAME'/
000021         000          KPUN=1                                                             00000140
000022         000          MAXT=0                                                             00000150
000023         000          IZQ=0
000024         000          READ (5 ,6,END=100) TITLE                                          00000160
000025         000          READ(5,LIST)
000026         000    C     NVBLES=NUMBER OF VARIABLES
000027         000    C     NSAMPL=NUMBER OF OBSERVATIONS
000028         000    C     INP=SWITCH FOR READ UNIT
000029         000    C     NTYP=NUMBER OF HYPOTHESIZED TYPES (IN SEQUFNCE)
000030         000    C     NOR=SWITCH FOR COV MATRICES (0 FOR COMMON, 1 FOR DIFFERENT)
000031         000    C     ENMIN=MINIMUM NUMBER OF POINTS IN A CLUSTER
000032         000    C     HTEST=PROBABILITY LEVEL FOR CHI SQUARE TEST
000033         000    C     MAXAV=SIZE OF SUB-SAMPLE USED TO MAKE INITIAL ESTIMATES
000034         000    C     IPRINT=SWITCH FOR PRINTING (1 FOR PRINTING AT EACH ITERATION)
000035         000    C     LTIME=TIME LIMIT IN MINUTES,IDEALLY 1 MIN LESS THAN JOB LIMIT
000036         000    C     ITERX=MAXIMUM NUMBER OF ITERATIONS
000037         000    C     ADDITIONAL COMMENTS ARE TO BE FOUND IN MAIN
000038         000    C
000039         000    C     IHIER=SWITCH FOR DISPLAYING HIERARCHICAL GROUPING (=1 FOR
000040         000    C     PRINTING THIRD AND FINAL ITERATION RESULT)
000041         000    C     MERGE=SWITCH FOR DISPLAY OF MERGE DETAILS.  THESE ARE IDENTICAL
000042         000    C     TO FINAL HIERARCHICAL GROUPING (=1 FOR PRINTING)
000043         000    C     IGEN=SWITCH FOR PRINTING EIGENVALUES AND EIGENVECTORS (=1 FOR
000044         000    C     PRINTING)
000045         000    C     MTIME=SWITCH FOR PRINTING TIME EXPENDED IN PRELIMINARY GROUPING
000046         000    C     AND IN DISCRIMINANT ANALYSIS (=1 FOR PRINTING)
000047         000    C     NTIME=SWITCH TO CUT OFF COMPUTATIONS AND PUNCH A DECK OF INPUT
000048         000    C     STATISTICS WHEN TIME IS CRITICAL (=1 FOR PUNCHING)
```

```
000    C      IHIS=SWITCH FOR CHI-SQUARE NORMALITY TEST AND HISTOGRAM FOR EACH
000    C      CLUSTER (=1 FOR TESTING)
000    C      IOVER=SWITCH FOR OVERRIDE OF ABORT TO TEST ADDITIONAL CLUSTER
000    C      (=1 FOR OVERRIDE)
000    C      IMAP=SWITCH FOR STORING INFORMATION TO BE USED IN ELLIPSE-PLOTTING
000    C      (=1 FOR STORING)
000    C      IBACK=SWITCH FOR READING STORED VALUES OF SCALING TERMS FROM
000    C      NORMIX-PREP. FOR PROFILES OR FOR STATISTICS ON ORIGINAL DATA
000    C      (=1 FOR BACKTRANSFORMING)
000    C      IUNITS=SWITCH FOR PRINTING NAME AND UNITS FOR EACH VARIABLE,
000    C      INSTEAD OF VARIABLE NUMBER (=1 FOR PRINTING)
000    C      IPROF=SWITCH FOR COMPUTING AND STORING INFORMATION TO BE USED IN
000    C      PROFILE PLOT OF CLUSTERING RESULTS (=1 FOR PROFILE)
000    C      IPOL=IDENTIFIERS OF VARIATES TO BE BACKTRANSFORMED IN PREPARATION
000    C      FOR PROFILE CHART AND/OR TAPLE OF STATISTICS FOR ORIGINAL DATA
000    C      POWER=EXPONENT APPLIED TO EACH DATUM IN NORMIX-PREP.
000    C      THE RECIPROCAL OF POWER IS USED TO BACKTRANSFORM IN PROFILE
000    C      NWIND=SWITCH FOR BACKTRANSFORMING WIND COMPONENTS TO SPED?ED AND
000    C      DIRECTION
000    C      IWIND=IDENTIFIER OF WIND VARIATES TO BE BACKTRANSFORMED
000    C
000           WRITE(6,LIST)
000           IAA=0
001           IP=0
000           IAB=0
001           CALL TABLE(NVBLES,IAA,IP,IAB)
000           READ 6,FMT
000           IF(INP.EQ.0) INP=5                                    00000190
000           IF (LTIME.EQ.0) LTIME=600                             00000200
000           NORMAP=(NOR.NE.1)                                     00000210
000           IF(ENMIN.EQ. 0.000) ENMIN=NVBLES+1                    00000220
000           IF (ITERX.EQ.0) ITERX=300                             00000230
000           DO 2 I=1,14                                           00000240
000           IF (NTYP(I).NE.0) GO TO 2                             00000250
000           DO 1 J=I,14                                           00000260
000           IF (NTYP(J).EQ.0) GO TO 1                             00000270
000           NTYP(I)=NTYP(J)                                       00000280
000           NTYP(J)=0                                             0000029
000           GO TO 2                                               00000300
000    1      CONTINUE                                              00000310
000           MAXT=I-1                                              00000320
000           GO TO 3                                               00000330
000    2      CONTINUE                                              00000340
000           MAXT=14                                               00000350
000    3      IF (MAXT.NE.0) GO TO 5                                00000360
000           MAXT=20                                               00000370
000           DO 4 I=1,20                                           00000380
000    4      NTYP(I)=I                                             00000390
000    5      MAXTYP=NTYP(MAXT)+1                                   00000400
000           NSEMI=NVBLES*(NVBLES+1)/2                             00000410
000           NPARAM=(NVBLES+1)*(NVBLES+2)/2                        00000420
```

```
000100     000           IF(NORMAP) NPARAM=NVBLES+1                                        00000430
000101     000           EN=NSAMPL                                                         00000440
000102     000           IF(MAXAV.EQ.0) MAXAV=150                                          00000450
000103     000           MAXAV=MIN0(MAXAV,NSAMPL)                                          00000460
000104     000           CALL TYTLE                                                        00000470
000105     000           ICOV = ISAME                                                      00000480
000106     000           IF (NOR.NE.0) ICOV = IDIF                                         00000490
000107     000           PRINT 10,  NVBLES,NSAMPL,ICOV                                     00000500
000108     000        10 FORMAT ('0NVBLES=',I3,'  NSAMPL=',I4,2X,A4,' COV MATRIX')         00000510
000109     000           NMIN = ENMIN                                                      00000520
000110     000           PRINT 11, NMIN,HTEST                                              00000530
000111     000        11 FORMAT (' MIN CLUSTER SIZE=',I3,'  HYPOTHESIS TEST=',F6.4)        00000540
000112     000           PRINT 12, MAXAV,LTIME,ITERX                                       00000550
000113     000        12 FORMAT (' INITIAL KMEANS=',I3,'  TIME LIMIT=',I3,                 00000560
000114     000          1'  ITER LIMIT=',I3)                                               00000570
000115     000           PRINT 13, (NTYP(L),L=1,14)                                        00000580
000116     000        13 FORMAT (' NO. OF TYPES=',14(1X,I2))                               00000590
000117     000           PRINT 15,FMT
000118     000    15     FORMAT(' FORMAT FOR READING DATA=  ',9A8)
000119     004    C
000120     004    C      IVALUE=SWITCH FOR PRELIMINARY ESTIMATES (=0 IF THERE
000121     004    C      ARE NONE TO READ IN)
000122     004    C      IMN=SWITCH FOR MEAN VALUE ESTIMATES (=0 IF THERE ARE
000123     004    C      NONE TO READ IN)
000124     004    C      ISTD=SWITCH FOR STANDARD DEVIATION ESTIMATES (=0 IF
000125     004    C      THERE ARE NONE TO READ IN)
000126     004    C      ICOR=SWITCH FOR CORRELATION COEFFICIENT ESTIMATES (=0
000127     004    C      IF THERE ARE NONE TO READ IN)
000128     000           READ(5,101,END=180)IVALUE,IMN,ISTD,ICOR
000129     000    101    FORMAT(4I5)
000130     002           RETURN
000131     000    180    WRITE(6,190)
000132     000    190    FORMAT(T2,'END OF FILE REACHED AT LABEL 101')
000133     000    6      FORMAT(9A8)
000134     003    100    NVBLES = -1                                                       00000650
000135     000           RETURN                                                            00000660
000136     000           END                                                               00000670

END ELT.
```

```
@ELT,L N.INITLE/CLUSTER
EL1007 SL73R1 02/25/80 16:09:04 (20,)
000001         6       SUBROUTINE INITLE (LAMBDA,MEAN,COVIN,R,S,X)                          00000010
000002         6       IMPLICIT REAL*8 (A-H,O-Z)                                            00000030
000003         6       REAL*8 LAMBDA,MEAN,MU,LIKLY
000004         6       LOGICAL NORMAP                                                       00000060
000005         6       COMMON/A/LIKLY,XLIMIT,NVBLES,NSAMPL,NTYPES,NSEMI,NPARAM,NORMAP,KX 00000070
000006         6       COMMON/B/CONVRG,CRITRN,RATIO,HTEST,OLDLIK,EN,BIGEST,THRESH,GUARD, 00000080
000007         6      1ENMIN,TITLE(18),FMT( 8),    MAXTYP,MAXAV,NCLUST,KSCRAT,INP,KPUN,KV,00000090
000008         6      2 LOT,LACK,LUCK,IPRINT,LTIME,MAXT,ITER,ITERX,LTYE,ITMC,NRANK,IVALUE00000100
000009         7      3,METHOD,NTYP(20)
000010         0       COMMON/F/IMAP,IUNITS,IPROF,IBACK,IPOL(20),SCAL(20),POWER,IWIND(2),
000011         0      *NWIND
000012         6       DIMENSION LAMBDA(1),MEAN(NVBLES,MAXTYP),COVIN(NSEMI,MAXTYP),        00000120
000013         6      1R(1),S(1),X(NVBLES,NSAMPL)
000014         6       NTYPES=0                                                             00000140
000015         6       ITER=0                                                               00000150
000016         6       LIKLY=0.0                                                            00000160
000017         6       DO 1 L=1,MAXTYP                                                      00000170
000018         6       LAMBDA(L)=0.0                                                        00000180
000019         6       DO 1 I=1,NVBLES                                                      00000190
000020         6 1     MEAN(I,L)=0.                                                         00000200
000021         6       NAXTYP=MAXTYP                                                        00000210
000022         6       IF(NORMAP) NAXTYP=3                                                  00000220
000023         6       DO 12 L=1,NAXTYP                                                     00000230
000024         6       DO 12 IJ=1,NSEMI                                                     00000240
000025         6 12    COVIN(IJ,L)=0.0D0                                                    00000250
000026         6       CALL TYTLE                                                           00000260
000027         6       IF(IMAP.NE.1)GO TO 21
000028         6       INR=11
000029         6       WRITE(INR,20)NVBLES,NSAMPL
000030         6 20    FORMAT(2I10)
000031         6 21    PRINT 90
000032         6       PRINT 95,(L,L=1,NVBLES)
000033         6 3     DO 4 J=1,NSAMPL                                                      00000290
000034         6 C     READ IN ALL THE DATA.                                                00000300
000035         6       READ (INP,FMT) (X(I,J),I=1,NVBLES)
000036         6       PRINT 99,J,(X(I,J),I=1,NVBLES)
000037         6       IF(IMAP.NE.1)GO TO 4
000038         6       WRITE(INR,30)(X(I,J),I=1,NVBLES)
000039         6 30    FORMAT(6E12.5,/,6E12.5)
000040         8 4     IF(J.EQ.NSAMPL.AND.IBACK.GT.0)READ(INP,FMT)(SCAL(I),I=1,NVBLES)
000041         6 40    DO 5 K=1,NSAMPL
000042         6       IJ=0                                                                 00000360
000043         6       DO 5 I=1,NVBLES                                                      00000370
000044         6       MEAN(I,1)=MEAN(I,1)+X(I,K)                                           00000380
000045         6       DO 5 J=I,NVBLES                                                      00000390
000046         6       IJ=IJ+1                                                              00000400
000047         6 5     COVIN(IJ,1)=COVIN(IJ,1)+X(I,K)*X(J,K)                                00000410
000048         6       IJ=0                                                                 00000420
000049         6       DO 7 I=1,NVBLES                                                      00000430
```

```
000050        6         DO 6 J=I,NVBLES                                                  00000440
000051        6         IJ=IJ+1                                                          00000450
000052        6 6       COVIN(IJ,1)=(COVIN(IJ,1)*EN-MEAN(I,1)*MFAN(J,1))/(EN*EN)         00000460
000053        6 7       MEAN(I,1)=MEAN(I,1)/EN                                           00000470
000054        6         RETURN                                                           00000480
000055        6    90   FORMAT (6X,'INPUT VARIABLES')                                    00000510
000056        6    95   FORMAT (1X,'SEQ',(12I10))                                        00000520
000057        6    99   FORMAT (1X,I6,12F10.4,/,(5X,12F10.4))                              00000530
000058        6         END                                                              00000540

END ELT.
```

```
@ELT,L N.KMEAN/CLUSTER
ELT007 SL73R1 02/25/80 16:09:05 (44,)
 0       SUBROUTINE KMEAN(AV,WT,A,BEST,MEMBER,KGRP,LGRP,KPATH,N,NS,NV,NTI) 00000010
 0 C     ADAPTED FROM WARD'S HIERARCHICAL GROUPING PROGRAM                 00000020
 0 C     BY JOHN H. WOLFE                    JUNE 10, 1970                  00000030
 0 C                                                                        00000040
 0 C     THIS ROUTINE COMBINES MACQUEEN'S KMEANS CLUSTERING PROCEDURE WITH 00000050
 0 C     WARD'S HIERARCHICAL GROUPING.                                      00000060
 0 C          ***REFERENCES***                                              00000070
 0 C     MACQUEEN,J.   FIFTH BERKELEY SYMPOSIUM(1965)                      00000080
 0 C     WARD,J.  PERSUB REFERENCE MANUAL(PRL-TR-67-3) (1967)              00000090
 0 C     DEFINITIONS:                                                       00000100
 0 C        N = MAXIMUM NUMBER OF GROUPS                                    00000110
 0 C        NS= NUMBER OF POINTS TO BE GROUPED.                             00000120
 0 C        NV= NUMBER OF VARIABLES.                                        00000130
 0 C        NTI=LOGICAL UNIT CONTAINING DATA TO BE CLUSTERED.               00000140
 0 C        WT(I)=NUMBER OF POINTS IN CLUSTER I.                            00000150
 0 C        AV(J,I)=MEAN OF VARIABLES J IN CLUSTER I.                       00000160
 0 C        A =  N*(N+1)/2 WORDS OF WORKING STORAGE USED BY THE PROGRAM    00000170
 0 C                                                                        00000180
 0 C         ON RETURN FROM THIS ROUTINE, KPATH WILL BE A LIST OF POINTS.  00000190
 0 C     KPATH(I) WILL BE NEGATIVE FOR THE FIRST MEMBER OF ONE OF THE N    00000200
 0 C     GROUPS AND KPATH(I+1),KPATH(I+2), ETC., WILL HAVE THE REMAINING   00000210
 0 C     MEMBERS IN THAT GROUP.  MEMBER(J) WILL BE THE NUMBER OF GROUPS    00000220
 0 C     LEFT AFTER THE J-TH GROUP IS MERGED WITH ANOTHER GROUP.           00000230
 0 C                                                                        00000240
 0 C     THIS ROUTINE EXPECTS THAT LOGICAL UNIT NTI CONTAINS THE DATA TO   00000250
 0 C     BE CLUSTERED.  THE DATA SHOULD HAVE BEEN PROPERLY SCALED OR       00000260
 0 C     CONVERTED TO FACTOR SCORES  AND WRITTEN AS UNFORMATTED (BINARY)   00000270
 0 C     RECORDS.                                                           00000280
 0 C                                                                        00000290
 0       IMPLICIT REAL*8 (A-H,O-Z)
 0       COMMON/D/IHIER,MERGE,IGEN,IZQ,IHIS
 0       DIMENSION A(1),KGRP(1),KPATH(1),WT(1),MEMBER(1),BEST(1),          00000300
 0      1AV(NV,N),LGRP(1)                                                   00000310
 0       DIMENSION BAR(5),GRAM(10)
 0       DATA BAR/8H . . . .,8H X . . .,8H X X . .,8H X X X .,8H X X X X/ 00000340
 0 C                                                                        00000350
 0 C     THE PROGRAM FIRST PERFORMS A KMEANS ANALYSIS AND THEN              00000360
 0 C     HIERARCHICALLY GROUPS THE GROUPS.                                  00000370
 0 C  1. INITIALLY, EACH OF THE FIRST N DATA POINTS IS A SEPARATE CLUSTER. 00000380
 0 C  2. THE INTER-GROUP DISTANCES, A, ARE COMPUTED.                       00000390
 0 C  3. THE NEAREST TWO GROUPS ARE COMBINED.                              00000400
 0 C  4. A NEW DATA POINT IS READ INTO THE SPACE LEFT BY THE MERGER.       00000410
 0 C  5. THE DISTANCES OF ALL THE CLUSTERS TO THIS NEW POINT ARE COMPUTED. 00000420
 0 C  6. GO BACK TO STEP 3 UNTIL ALL THE DATA HAS BEEN READ.               00000430
 0 C  7. HIERARCHICALLY  GROUP THE N GROUPS.                               00000440
 0 C                                                                        00000450
 0 C     INITIALIZE                                                         00000460
 0       NA = (N * (N+1))/2                                                 00000470
```

```
000049     0          NTWO=N+N                                                        00000480
000050     0          NONE=N-1                                                        00000490
000051     0          IF(NTI.NE.0) REWIND NTI                                         00000500
000052     0          KS=N                                                            00000510
000053     0          NUMG=N                                                          00000520
000054     0          DO 10  I=1,N                                                    00000530
000055     0          IF(NTI.NE.0) READ (NTI) (AV(K,I),K=1,NV)                        00000540
000056     0          LGRP(I)=I                                                       00000550
000057     0          KGRP(I)=I+1                                                     00000560
000058     0          BEST(I)=1.0D35
000059     0          WT(I)=1.0
000060     0 10       CONTINUE
000061     0          KGRP(N)=0                                                       00000590
000062     0          DO 15 I=1,NS                                                    00000600
000063     0  15      KPATH(I)=0                                                      00000610
000064     0 C        A GROUP IS IDENTIFIED BY THE LEAST INDEX OF ITS MEMBERS         00000620
000065     0 C        IF I IS A MEMBER OF A GROUP, KPATH(I) IS ANOTHER MEMBER.        00000630
000066     0 C        IF I IS THE BEGINNING OF A GROUP, KGRP(I) BEGINS THE NEXT GROUP. 00000640
000067     0 C        LGRP(I)= THE LEADING MEMBER OF GROUP I                          00000650
000068     0 C        A DIAGONAL ELEMENT OF A IS THE WITHIN-GROUP DISPERSION.         00000660
000069     0 C        THE VALUE OF A IN ROW I,COLUMN J IS THE DISPERSION WHICH WOULD  00000670
000070     0 C        RESULT FROM COMBINING GROUPS I AND J .                          00000680
000071     0 C        COMPUTE DI  BLOCK 2                                             00000690
000072     0          IJ=0                                                            00000700
000073     0          DO 40 I=1,NONE                                                  00000710
000074     0          IJ=IJ+1                                                         00000720
000075     0          A(IJ)=0.0                                                       00000730
000076     0          DO 30 J=I,NONE                                                  00000740
000077     0          IJ=IJ+1                                                         00000750
000078     0          A(IJ)=0.0                                                       00000760
000079     0          DO 20 K=1,NV                                                    00000770
000080     0  20      A(IJ)=A(IJ)+(AV(K,I)-AV(K,J+1))**2                              00000780
000081     0          A(IJ)=A(IJ)/2.0                                                 00000790
000082     0          DIT=A(IJ)                                                       00000800
000083     0          IF(DIT.GE.BEST(I)) GO TO 30                                     00000810
000084     0          BEST(I)=DIT                                                     00000820
000085     0          MEMBER(I)=J+1                                                   00000830
000086     0  30      CONTINUE                                                        00000840
000087     0  40      CONTINUE                                                        00000850
000088     0          A(NA)=0.0                                                       00000860
000089     0  50      I=1                                                             00000870
000090     0          DI=1.0D35                                                       00000880
000091     0          DO 60 J=1,NUMG                                                  00000890
000092     0          IF(BEST(I).GE.DI) GO TO 60                                      00000900
000093     0          DI=BEST(I)                                                      00000910
000094     0          IDS=I                                                           00000920
000095     0  60      I=KGRP(I)                                                       00000930
000096     0  70      JDS=MEMBER(IDS)                                                 00000940
000097     0          IF(JDS.GT.IDS) GO TO 100                                        00000950
000098     0          IDS=JDS                                                         00000960
```

```
0        GO TO 70
0 C      UPDATE KGRP AND KPATH  BLOCK 3
0 100    I=IDS
0        NUMG=NUMG-1
0        BEST(JDS)=DI
0 110    IF(KGRP(I)-JDS)120,130,120
0 120    I=KGRP(I)
0        GO TO 110
0 130    KGRP(I)=KGRP(JDS)
0        KGRP(JDS)=-NUMG
0        I=LGRP(IDS)
0 140    IF(KPATH(I))150,160,150
0 150    I=KPATH(I)
0        GO TO 140
0 160    KPATH(I)=LGRP(JDS)
0 C      UPDATE HYP MATRIX BLOCK 4
0        I=1
0        BEST(IDS)=1.0D35
0        L =((NTWO-IDS)*(IDS-1))/2+IDS
0        K =((NTWO-JDS)*(JDS-1))/2+JDS
0        M=L+JDS-IDS
0        PLOP=A(M)
0        TSUM1=A(M)*(WT(IDS)+WT(JDS))
0        TSUM3= (A(L)* WT(IDS))+ (A(K)*WT(JDS))
0        A(L)=PLOP
0        DO 350 IM=1,NUMG
0        II=((NTWO-I  )*(I   -1))/2+I
0        IF(I-IDS)210,350,240
0 210    M=II+IDS-I
0        GO TO 250
0 240    M=L+I-IDS
0 250    IF(I-JDS)260,270,270
0 260    M1=II+JDS-I
0        GO TO 280
0 270    M1=K+I-JDS
0 280    A(M)=(A(M1)*(WT(I)+WT(JDS))+ (
0       1A(M)*(WT(I)+WT(IDS)))+  (TSUM1
0       2-TSUM3-(A(II)*WT(I))))/(WT(I)+WT(JDS)+WT(IDS) )
0        DIT=A(M)-PLOP-A(II)
0        IF(DIT.GE.BEST(IDS)) GO TO 290
0        BEST(IDS)=DIT
0        MEMBER(IDS)=I
0 290    IF(MEMBER(I).NE.IDS.AND.MEMBER(I).NE.JDS) GO TO 340
0        BEST(I)=1.0D35
0        J=I
0 300    J=KGRP(J)
0        IF(J.LE.0) GO TO 350
0        JJ=((NTWO-J)*(J-1))/2+J
0        IJ=II+J-I
0        DOT=A(IJ)-A(II)-A(JJ)
0        IF(DOT.GE.BEST(I)) GO TO 300
```

```
0           BEST(I)=DOT
0           MEMBER(I)=J
0           GO TO 300
0 340       IF(DIT.GE.BEST(I)) GO TO 350
0           BEST(I)=DIT
0           MEMBER(I)=IDS
0 350       I=KGRP(I)
0           WTS=WT(IDS)
0           WT(IDS)=WT(IDS)+WT(JDS)
0           IF(KS.EQ.NS) GO TO 359
0           KS=KS+1
0           NUMG=N
0           LGRP(JDS)=KS
0           KGRP(JDS-1)=JDS
0           KGRP(JDS)= JDS+1
0           KGRP(N)=0
0           DO 351 I=1,NV
0     351 AV(I,IDS)=(WTS*AV(I,IDS)+WT(JDS)*AV(I,JDS))/WT(IDS)
0           READ(NTI) (AV(I,JDS),I=1,NV)
0           WT(JDS)=1.0
0           BEST(JDS)=1.0D35
0           M=JDS
0           JADD=N
0           JJ=1
0           DO 355 J=1,N
0           DIT=A(JJ)
0           IF(J.EQ.JDS) GO TO 354
0           DO 352 I=1,NV
0     352 DIT =DIT +(AV(I,JDS)-AV(I,J))**2
0           A(M)=DIT *WT(J   )/(WT(J   )+1.0)
0           DIT=A(M)-A(JJ)
0           IF(J.GT. JDS) GO TO 353
0           IF(DIT .GE. BEST(J)) GO TO 354
0           BEST(J)=DIT
0           MEMBER(J)=JDS
0           GO TO 354
0 353       IF(DIT .GE. BEST(JDS)) GO TO 354
0           BEST(JDS)=J
0           MEMBER(JDS)=J
0           BEST(JDS)=DIT
0  354      JADD=JADD-1
0           IF(J.GE.JDS) JADD=1
0           JJ=JJ+N-J+1
0     355 M=M+JADD
0 359       IF(NUMG-1) 360,360,50
0 C         OUT PUT  BLOCK 5
0 360       LA=0
0           DO 361 I=1,N
0           L=LGRP(I)
0           IF(KPATH(L).EQ.0) LA=I
```

```
000200         0 361    KPATH(L)=-KPATH(L)                                          00001980
000201         0        CALL RELIST(KPATH,NSAMPL)                                   00001990
000202         0        K=0                                                         00002000
000203         0        DO 362 I=2,NS                                               00002010
000204         0        IF(KPATH(I).GE.0) GO TO 362                                 00002020
000205         0        K=K+1                                                       00002030
000206         0        DO 363 L=1,N                                                00002040
000207         0        IF(LGRP(L).EQ. KPATH(I-1)) GO TO 364                        00002050
000208         0 363    CONTINUE                                                    00002060
000209         0 364    MEMBER(K)=L                                                 00002070
000210         0        KPATH(I-1)=-KPATH(I-1)                                      00002080
000211         0        KPATH(I)=-KPATH(I)                                          00002090
000212         0        WT(K)=0.0                                                   00002100
000213         0 362    WT(K)=WT(K)+1.0                                             00002110
000214         0        WT(N)=WT(N)+1.0                                             00002120
000215         0        IF(LA.EQ.0) GO TO 371                                       00002130
000216         0        KPATH(NS)=-KPATH(NS)                                        00002140
000217         0        MEMBER(N)=LA                                                00002150
000218         0        WT(N)=1.0                                                   00002160
000219         0        DO 370 K=1,N                                                00002170
000220         0        L=MEMBER(K)                                                 00002180
000221         0 366    IF(L-K) 367,370,368                                         00002190
000222         0 367    L=MEMBER(L)                                                 00002200
000223         0        GO TO 366                                                   00002210
000224         0 368    DO 369 J=1,NV                                               00002220
000225         0        TEMP=AV(J,K)                                                00002230
000226         0        AV(J,K)=AV(J,L)                                             00002240
000227         0 369    AV(J,L)=TEMP                                                00002250
000228         0 370    CONTINUE                                                    00002260
000229         0 371    DO 372 L=1,N                                                00002270
000230         0        K=MEMBER(L)                                                 00002280
000231         0        J=-KGRP(K)                                                  00002290
000232         0        MEMBER(L)=J                                                 00002300
000233         0 372    LGRP(J+1)=L                                                 00002310
000234         0 C      MEMBER(L)= STAGE OF GROUP L                                 00002320
000235         0 C      LGRP(J)= GROUP WHOSE STAGE IS J-1                           00002330
000236         0        L=0                                                         00002340
000237         0        DO 373 K=1,NS                                               00002350
000238         0        IF(KPATH(K).GT.0) GO TO 373                                 00002360
000239         0        L=L+1                                                       00002370
000240         0        KGRP(L)=K                                                   00002380
000241         0 373    CONTINUE                                                    00002390
000242         0 C      KGRP(L)= THE POSITION IN KPATH WHICH BEGINS GROUP L         00002400
000243         0 365    IZQ=IZQ+1
000244         0        DO 430 L=1,N,40                                             00002410
000245         0        L2=MINO(L+35,N)                                             00002420
000246         3        IF(IZQ.EQ.3.AND.IHIER.EQ.1)PRINT 1000,(J,J=L,L2,5)          00002430
000247         0        M=0                                                         00002440
000248         0        DO 420 I=1,NS                                               00002450
000249         0        MEM=N+1                                                     00002460
```

```
000250         0         J=KPATH(I)                                                     00002470
000251         0         IF(J.GT.0) GO TO 379                                           00002480
000252         0         M=M+1                                                          00002490
000253         0         MEM=MEMBER(M)                                                  00002500
000254         0         J=-J                                                           00002510
000255         0 379     LK4=MAX0(0,MEM -L+1)                                           00002520
000256         0         LM4=LK474                                                      00002530
000257         0         LN4=LK4-4*LM4                                                  00002540
000258         0         DO 410 K=1,10                                                  00002550
000259         0         IF(K-LM4-1) 380,390,400                                        00002560
000260         0 380     GRAM(K)=BAR(5)                                                 00002570
000261         0         GO TO 410                                                      00002580
000262         0 390     GRAM(K)=BAR(LN4+1)                                             00002590
000263         0         GO TO 410                                                      00002600
000264         0 400     GRAM(K)=BAR(1)                                                 00002610
000265         0 410     CONTINUE                                                       00002620
000266         3         IF(IZQ.EQ.3.AND.IHIER.EQ.1)PRINT 1120,I,KPATH(I),M,MEM,GRAM
000267         0 420     CONTINUE
000268         0 430     CONTINUE                                                       00002640
000269         1 1000    FORMAT(1H1,51X,'HIERARCHICAL GROUPING'//66X,'NUMBER OF GROUPS'/3X 00002650
000270         0        1,'RANK  ITEM  KMEANS STAGE' 5X,8I10)                           00002660
000271         0 1120   FORMAT(2I6,2I7,14X,10A8)                                        00002670
000272         0         RETURN                                                         00002680
000273         0         END

END ELT.
```

```
@ELT,L N.MAPPER/CLUSTER
EL1007 SL73R1 02/25/80 16:09:07 (12,)
000001          8       SUBROUTINE MAPPER(MATRIX,R,B)                                   00000010
000002          8       IMPLICIT REAL*8 (A-H,O-Z)                                       00000020
000003          8       REAL*8 LAMBDA,MEAN,MU,LIKLY
000004          8       LOGICAL NORMAP                                                  00000040
000005          8       COMMON/A/LIKLY,XLIMIT,NVBLES,NSAMPL,NTYPES,NSEMI,NPARAM,NORMAP,KX 00000050
000006          8       COMMON/B/CONVRG,CRITRN,RATIO,HTEST,OLDLIK,EN,BIGEST,THRESH,GUARD, 00000060
000007          9      1ENMIN,TITLE(18),FMT( 8),    MAXTYP,MAXAV,NCLUST,KSCRAT,INP,KPUN,KV,00000070
000008          8      2 LOT,LACK,LUCK,IPRINT,LTIME,MAXT,ITER,ITERX,LTYE,ITMC,NRANK,IVALUE00000080
000009          2      3,METHOD,NTYP(20)
000010          8       DIMENSION LOOKUP(20), MATRIX(120,50),R(NRANK),B(21)             00000100
000011          8       DATA  LOOKUP/1H1,1H2,1H3,1H4,1H5,1H6,1H7,1H8,1H9,1HA,1HB,1HC,1HD, 00000110
000012          8      1 1HE,1HF,1HG,1HH,1HI,1HJ,1HK /,KBLK/1H  /,KPER/1H. /,KSTR/1H* /   00000120
000013          8 C     THIS ROUTINE READS THE TAPE OF DISCRIMINANT SCORES AND PRODUCES A 00000130
000014          8 C     SEMI-GRAPHICAL DISPLAY ON THE PRINTOUT.                         00000140
000015          8       KC=3                                                            00000150
000016          8       IF(NRANK.LE.1) RETURN                                           00000160
000017          8       LTYPES=NTYPES-1                                                 00000170
000018          8       DO 10 I=1,2                                                     00000180
000019          8       I1=I+1                                                          00000190
000020          8       IF(I1.GT.NRANK) GO TO 11                                        00000200
000021          8       DO 10 J=I1,NRANK                                                00000210
000022          8       DO 2 M=1,48                                                     00000220
000023          8       DO 2 N=1,116                                                    00000230
000024          8 2     MATRIX(N,M)=KBLK                                                00000240
000025          8       DO 3 M=1,48                                                     00000250
000026          8 3     MATRIX(1,M)=KPER                                                00000260
000027          8       DO 4 N=1,116                                                    00000270
000028          8 4     MATRIX(N,47)=KPER                                               00000280
000029          8       REWIND KC                                                       00000290
000030          8       DO 5 K=1,NSAMPL                                                 00000300
000031          8       READ(KC)L,KEY,(R(II),II=1,NRANK)                                00000310
000032          8       M=5.0*R(I)+58.                                                  00000320
000033          8       N=-3.0*R(J)+23.                                                 00000330
000034          8       M= MAXO(2,MINO(M,116))                                          00000340
000035          8       N= MAXO(2,MINO(N, 46))                                          00000350
000036          8       LIM=LOOKUP(KEY-1)                                               00000360
000037          8       IF (MATRIX(M,N).EQ.KBLK) GO TO 5                                00000370
000038          8       IF (MATRIX(M,N).NE.LIM) LIM=KSTR                                00000380
000039          8 5     MATRIX(M,N)=LIM                                                 00000390
000040          8       CALL TYTLE                                                      00000400
000041          8       PRINT 13, J,LTYPES                                              00000410
000042          8       LY=8                                                            00000420
000043          8       LB=1                                                            00000430
000044          8       DO 8 N=1,47                                                     00000440
000045          8       LB=LB+1                                                         00000450
000046          8       GO TO (7,7,6), LB                                               00000460
000047          8 6     LB=0                                                            00000470
000048          8       LY=LY-1                                                         00000480
000049          8       PRINT 14, LY,(MATRIX(M,N),M=1,116)                              00000490
```

```
                                                                                              00000500
000050      8        GO TO 8                                                                  00000510
000051      8 7      PRINT 15, (MATRIX(M,N),M=1,116)                                          00000520
000052      8 8      CONTINUE                                                                 00000530
000053      8        XD=-11.0                                                                 00000540
000054      8        DO 9 M=1,21                                                              00000550
000055      8        XD=XD+1.0                                                                00000560
000056      8 9      A(M)=XD                                                                  00000570
000057      8 10     PRINT 16, (B(M),M=1,21),I                                                00000580
000058      8 11     CONTINUE                                                                 00000590
000059      8        RETURN                                                                   00000600
000060      8 C                                                                               00000610
000061      8 12     FORMAT (6X,I4,10F7.2)                                                    00000620
000062      8 13     FORMAT (5X,21HDISCRIMINANT FUNCTIONI2,50X,16HNUMBER OF TYPES=I2)         00000630
000063      8 14     FORMAT (I4,116A1)                                                        00000640
000064      8 15     FORMAT (4X,116A1)                                                        00000650
000065      8 16     FORMAT (F12.0,20F5.0/45X,21HDISCRIMINANT FUNCTIONI2)                     00000660
000066      8        END

END ELT.
```

```
@ELT,L N.MOMENT/CLUSTER
ELT007 SL73R1 02/25/80 16:09:08 (14,)
000001        0         SUBROUTINE MOMENT (MEAN,COVIN,X,R,S,B,G,MU,V)
000002        0         IMPLICIT REAL*8 (A-H,O-Z)                                  00000020
000003        0         REAL*8 LAMBDA,MEAN,MU,LIKLY
000004        0         LOGICAL NORMAP                                             00000050
000005        0         COMMON/A/LIKLY,XLIMIT,NVBLES,NSAMPL,NTYPES,NSEMI,NPARAM,NORMAP,KX 00000060
000006        0         DIMENSION MEAN(1),COVIN(1),X(1),R(1),S(1)                  00000070
000007        0        1,B(1),G(1),MU(1),V(1)                                      00000080
000008        0 C       THIS ROUTINE COMPUTES                                      00000090
000009        0 C       MU(1,L)= SUM OVER ALL INDIVIDUALS OF THEIR PROBABILITIES OF 00000100
000010        0 C       MEMBERSHIP IN TYPE L.                                      00000110
000011        0 C       MU(I+1,L)= SUM OVER ALL INDIVIDUALS OF THEIR X(I,K)        00000120
000012        0 C       TIMES THEIR PROBABILITY OF MEMBERSHIP IN TYPE L.           00000130
000013        0 C       MU(IJ,L) = SUM OVER ALL INDIVIDUALS OF THEIR X(I,K)*X(J,K) 00000140
000014        0 C       TIMES THEIR PROBABILITY OF MEMBERSHIP IN TYPE L.           00000150
000015        0 C       G(L)=PROBABILITY OF MEMBERSHIP IN TYPE L,                  00000160
000016        0 C                                                                  00000170
000017        0 C                                                                  00000180
000018        2 C       THIS SUBROUTINE AND SUBROUTINE DENSIT CONTAINS THE MOST DEEPLY 00000190
000019        2 C       NESTED LOOPS OF THE ENTIRE NORMIX PROGRAM.
000020        2 C       THE FOLLOWING TRICKS ARE USED TO IMPROVE OBJECT CODE
000021        2 C       EFFICIENCY
000022        0 C       --- THE ROUTINE COMPUTES ITS OWN INDEXES FOR THE           00000220
000023        0 C       DOUBLY-INDEXED ARRAYS INSTEAD OF DEPENDING ON THE COMPILER. 00000230
000024        0 C       X(IK)   IS   X(I,K)                                        00000240
000025        0 C       MEAN(IL)   IS MEAN (I,L)                                   00000250
000026        0 C       COVIN(M) IS COVIN(I,J,L)                                   00000260
000027        0 C       MU(IJ)  IS MU(I,J,L)                                       00000270
000028        0         DO 12 K=1,NSAMPL                                           00000280
000029        0               CALL DENSIT(MEAN,COVIN,X,R,S,B,V,G,K)                00000290
000030        0         IF(NORMAP) GO TO 44                                        00000300
000031        0         IJ=NPARAM                                                  00000310
000032        0         DO 11 L=2,NTYPES                                           00000320
000033        0         PM=G(L)                                                    00000330
000034        0         IA=IJ+1                                                    00000340
000035        0         IJ=IA+NVBLES                                               00000350
000036        0         MU(IA)=PM+MU(IA)                                           00000360
000037        0         DO 11 I=1,NVBLES                                           00000370
000038        0         Z=R(I)*PM                                                  00000380
000039        0         IA=IA+1                                                    00000390
000040        0         MU(IA)=Z+MU(IA)                                            00000400
000041        0         DO 11 J=I,NVBLES                                           00000410
000042        0         IJ=IJ+1                                                    00000420
000043        0 11      MU(IJ)=MU(IJ)+Z*R(J)                                       00000430
000044        0         GO TO 12                                                   00000440
000045        0 44      IA=NPARAM                                                  00000450
000046        0         DO 46 L=2,NTYPES                                           00000460
000047        0         PM=G(L)                                                    00000470
000048        0         IA=IA+1                                                    00000480
000049        0         MU(IA)=PM+MU(IA)                                           00000490
```

```
000050      0       DO 46 I=1,NVBLES                                                    00000500
000051      0       IA=IA+1                                                             00000510
000052      0 46    MU(IA)=R(I)*PM+MU(IA)                                               00000520
000053      0 12    CONTINUE                                                            00000530
000054      0       RETURN                                                              00000540
000055      0       END                                                                 00000550
```

```
@ELT,L N.NORMIX/CLUSTER
EL1007 SL73R1 02/25/80 16:09:09 (32,)
000001       8        SUBROUTINE NORMIX(LAMBDA,MEAN,COVIN,MU,OLAM,OMEAN,V,T,W,BMAT,R,S  00000010
000002       8       1 ,B,G,AV,DENSE,MEMBER,X,KPATH)                                    00000020
000003       8        IMPLICIT REAL*8 (A-H,O-Z)                                         00000030
000004       8        REAL*8 LAMBDA,MEAN,MU,LIKLY                                       00000040
000005       8        INTEGER G,B
000006       8        LOGICAL NORMAP                                                    00000060
000007       8        COMMON/A/LIKLY,XLIMIT,NVBLES,NSAMPL,NTYPES,NSEMI,NPARAM,NORMAP,KX 00000070
000008       8        COMMON/B/CONVRG,CRITRN,RATIO,HTEST,OLDLIK,EN,BIGEST,THRESH,GUARD, 00000080
000009       8       1ENMIN,TITLE(18),FMT( 8),    MAXTYP,MAXAV,NCLUST,KSCRAT,INP,KPUN,KV,00000090
000010       8       2 LOT,LACK,LUCK,IPRINT,LTIME,MAXT,ITER,ITERX,LTYE,ITMC,NRANK,IVALUE00000100
000011       1       3,METHOD,NTYP(20)
000012       9        COMMON/E/MTIME,NTIME,IOVER
000013       8        DIMENSION LAMBDA( 1),MEAN( 1),COVIN( 1),R( 1),S( 1),B( 1),G( 1),  00000120
000014       8       1MU( 1),AV( 1),OLAM(  1),OMEAN(  1),DENSE(  1),X(NVBLES,NSAMPL),   00000130
000015       8       2V(  1),T(  1),W(  1),BMAT(  1),MEMBER(  1),KPATH(  1)             00000140
000016       8 C                                                                        00000150
000017       8 C      THIS IS THE CONTROLLING PROGRAM WHICH CALLS THE OTHER SUBROUTINES.00000160
000018       8 C                                                                        00000170
000019       8 C      NOTICE THAT, AS PRESENTLY PROGRAMMED, ALL DATA MUST BE IN CORE AND00000180
000020       8 C      STORED IN THE ARRAY X. IT WOULD BE RELATIVELY EASY TO REPROGRAM TO00000190
000021       8 C      HANDLE MUCH LARGER PROBLEMS BY READING X ONE RECORD AT A TIME FROM00000200
000022       8 C      DISK EACH TIME INITLE, HISTOG, MOMENT, AND ASSIGN ARE CALLED. ONE 00000210
000023       8 C      SHOULD BE CAREFUL ABOUT THIS, SINCE X IS USED AS A WORKING STORAGE00000220
000024       8 C      AREA IN KMEAN AND MAPPER. ALSO, THIS METHOD WILL SLOW THE PROGRAM 00000230
000025       8 C      BY ABOUT 20 PERCENT.                                              00000240
000026       8 C                                                                        00000250
000027       8        CONVRG=1.0D-4                                                     00000260
000028       8        KSCRAT=4                                                          00000270
000029       8        REWIND KSCRAT                                                     00000280
000030       8        CALL ITIME (ITMA)                                                 00000290
000031       8 C      ITIME IS A ROUTINE WHICH RETURNS THE CLOCK VALUE IN 100-THS OF A  00000300
000032       8 C      SECOND.                                                           00000310
000033       8 C      LTIME IS THE TIME LIMIT IN MINUTES.                               00000320
000034       8         ITMA=6000*LTIME+ITMA                                             00000330
000035       8        CALL INITLE (LAMBDA,MEAN,COVIN,R,S,X)                             00000340
000036       8              CALL HISTOG(MEAN,AV,DENSE,X,R,S,G,B,MU,COVIN,T,MEMBER,KPATH)00000350
000037       8        CALL ITIME(ITMB)                                                  00000370
000038       8        TMB=ITMB/100.0                                                    00000380
000039       8        IF(MTIME.EQ.1)PRINT 15,TMB                                        00000390
000040       8        DO 8 L=1,MAXT                                                     00000400
000041       8        LTYE=L                                                            00000410
000042       8        ITER=0                                                            00000420
000043       8        METHOD=10 *NVBLES                                                 00000430
000044       8        NTYPES=NTYP(L)+1                                                  00000440
000045       8              CALL REVALU (LAMBDA,MEAN,COVIN,AV,DENSE,MEMBER,R,S)         00000450
000046       2        CALL OUTPUT (LAMBDA,MEAN,COVIN,R,S,W,T,X)                         00000470
000047       8        OLDLIK=-1.0D08                                                    00000480
000048       8        CALL TYTLE                                                        00000490
000049       8        LDIV=0                                                            00000500
```

```
8          CALL ITIME(ITME)                                                    00000510
8          DO 5 ITER=1,ITERX                                                   00000520
8                CALL AITKEN (LAMBDA,MEAN,COVIN,X,R,S,B,V,G,MU,OLAM,OMEAN)     00000530
8 C   CHECK FOR ZERO PIVOT                                                     00000550
8          DO 40 LT = 1,NVBLES                                                 00000560
8          IF (G(LT).EQ.1.) GO TO 41                                           00000570
8   40     CONTINUE                                                            00000580
8          FLIK=LIKLY-OLDLIK                                                   00000590
8          DLIK=DABS(ELIK)                                                     00000600
8          IF (ELIK.GT.-1.0D-5) GO TO 1                                        00000610
8          LDIV=LDIV+1                                                         00000620
8          IF (LDIV.LT.9) GO TO 2                                              00000630
8 C        IF THE LIKELIHOOD FAILS 9 TIMES TO EXCEED THE MAXIMUM LIKELIHOOD    00000640
8 C        OF PRIOR ITERATIONS,TERMINATE JOB.                                  00000650
8          PRINT 10                                                            00000660
8          IF (DLIK.GT.CONVRG) ITMC=10                                         00000670
8          GO TO 7                                                             00000680
8 1        OLDLIK=LIKLY                                                        00000690
8          LDIV=0                                                              00000700
8 C        TEST WHETHER THE CONVERGENCE CRITERION IS MET.                      00000710
8          IF (DLIK-CONVRG) 7,7,2                                              00000720
8 2        IF (IPRINT.EQ.0) GO TO 3                                            00000730
0          CALL OUTPUT (LAMBDA,MEAN,COVIN,R,S,W,T,X)                           00000740
8          GO TO 4                                                             00000750
8 3        PRINT 11, ITER,NTYP(L),LIKLY                                        00000760
8 4        CONTINUE
8          CALL ITIME (ITMB)                                                   00000770
8          ITMC=ITMB-ITMA                                                      00000780
8          TMC=ITMC/100.                                                       00000790
8          IF(NTIME.NE.1)GO TO 5
8 C        STOP AND DUMP CURRENT PARAMETER ESTIMATES IF TIME LIMIT REACHED.    00000800
8          IF(TMC.LT.0)GO TO 5
8          PRINT 13, TMC                                                       00000820
8          GO TO 6                                                             00000830
8 5        CONTINUE                                                            00000840
8          GO TO 7                                                             00000850
8 6        CALL DUMPER (LAMBDA,MEAN,COVIN,R,S)                                 00000860
8 7        CONTINUE
8          CALL ITIME(ITMB)                                                    00000870
8          TME=(ITMB-ITME)/100.0                                               00000880
8          TMR=ITMB/100.                                                       00000890
8          IF(NTIME.EQ.1)PRINT 14,TME,TMR                                      00000900
2          CALL OUTPUT(LAMBDA,MEAN,COVIN,R,S,W,T,X)
8          IF(NTYPES.EQ.2) GO TO 20                                            00000920
8          NRANK=MINO(NTYPES-2,NVBLES,R)                                       00000930
8          CALL ASSIGN(LAMBDA,MEAN,COVIN,X,R,S,B,V,G,MU,T,W,BMAT)              00000940
8 C   CHECK FOR ZERO PIVOT                                                     00000960
8          DO 42 LT = 1,NVBLES                                                 00000970
8          IF (G(LT).EQ.1.) GO TO 41                                           00000980
8   42     CONTINUE                                                            00000990
```

```
8 C   CHECK FOR PROPORTION < .02 IN ANY GROUP
8         DO 30 LYY=2,NTYPES
8         IF (LAMBDA(LYY).LT..02) GO TO 31
8   30    CONTINUE
8         IF(NORMAP) CALL MAPPER(X,R,S)
8         REWIND KV
8         READ(KV)X
8 C       TEST WHETHER LIKELIHOOD IS SIGNIFICANTLY GREATER FOR MORE TYPES.
8 20      RATIO=LIKLY-PRELIK
8         PRELIK=LIKLY
8         NTYPES=NTYP(L)
8         NTYOLD=NTYP(L-1)
8         NDEGRE=(NPARAM-1)*(NTYPES-NTYOLD)*2
8         IF (L.LE.1) GO TO 8
8         CHISQ=2.0*RATIO*(EN-0.5-NVBLES-NTYPES/2.0)/EN
8         PCHISQ=CHISQP(CHISQ,NDEGRE)
8         CALL TYTLE
8         PRINT 12, NTYPES,NTYOLD,RATIO,NDEGRE,CHISQ,PCHISQ
8         IF(IOVER.EQ.1)GO TO 8
8         IF (PCHISQ-HTEST) 8,8,9
8 8       CONTINUE
8 9       CONTINUE
8         IF(NORMAP) END FILE KSCRAT
8         RETURN
8 C
8   31    LYY = LYY-1
8         PRINT 16,LYY
8         GO TO 9
8   41    PRINT 17
8         GO TO 9
8 C
8 10      FORMAT (/5X,45HJOB TERMINATED--ITERATIONS ARE NOT CONVERGING)
8 11      FORMAT (10H ITERATIONI4,10X,17HLOG LIKELIHOOD OFI3,23H TYPES IN TH
8        1IS SAMPLE =E18.8)
8 12      FORMAT (33H LOGARITHM OF LIKELIHOOD RATIO OF,I3,3H TO,I3,8H TYPES 
8        1=E18.8/16H CHI-SQUARE WITH,I4,20H DEGREES OF FREEDOM=,F10.2/32H PR
8        2OBABILITY OF NULL HYPOTHESIS=F11.8)
8 13      FORMAT (39H TIME LIMIT EXCEEDED. COMPUTATION TIME=F6.2,8H SECONDS)
8 14      FORMAT(//40X,'ITERATION TIME =' F12.2,' SECONDS.'/40X,'TOTAL TIME 
8        1USED =' F11.2,' SECONDS.')
8 15      FORMAT(//40X,'TIME IN HIERARCHICAL GROUPING =' F8.2,' SECONDS.')
8   16    FORMAT (/5X,'JOB TERMINATED--PROPORTION OF POPULATION OF TYPE'
8        X          ,I3,' LESS THAN .02')
8   17    FORMAT (/,5X,'JOB TERMINATED - PIVOT OF 0.0 IN SYMINV')
8         END

END ELT.
```

```
@ELT,L N.NROOT/CLUSTER
EL1007 SL73R1 02/25/80 16:09:11 (3,)
000001      000         SUBROUTINE NROOT (M,A,B,XL,X)                                   00000010
000002      000         REAL*8 DSQRT                                                    00000020
000003      000    C                                                                    00000030
000004      000    C    ................................................................00000040
000005      000    C                                                                    00000050
000006      000    C       SUBROUTINE NROOT                                             00000060
000007      000    C                                                                    00000070
000008      000    C       PURPOSE                                                      00000080
000009      000    C          COMPUTE EIGENVALUES AND EIGENVECTORS OF A REAL NONSYMMETRIC 00000090
000010      000    C          MATRIX OF THE FORM B-INVERSE TIMES A.  THIS SUBROUTINE IS  00000100
000011      000    C          NORMALLY CALLED BY SUBROUTINE CANOR IN PERFORMING A        00000110
000012      000    C          CANONICAL CORRELATION ANALYSIS.                           00000120
000013      000    C                                                                    00000130
000014      000    C       USAGE                                                        00000140
000015      000    C          CALL NROOT (M,A,B,XL,X)                                   00000150
000016      000    C                                                                    00000160
000017      000    C       DESCRIPTION OF PARAMETERS                                    00000170
000018      000    C          M  - ORDER OF SQUARE MATRICES A, B, AND X.                00000180
000019      000    C          A  - INPUT MATRIX (M X M).                                00000190
000020      000    C          B  - INPUT MATRIX (M X M).                                00000200
000021      000    C          XL - OUTPUT VECTOR OF LENGTH M CONTAINING EIGENVALUES OF   00000210
000022      000    C               B-INVERSE TIMES A.                                   00000220
000023      000    C          X  - OUTPUT MATRIX (M X M) CONTAINING EIGENVECTORS COLUMN- 00000230
000024      000    C               WISE.                                                00000240
000025      000    C                                                                    00000250
000026      000    C       REMARKS                                                      00000260
000027      000    C          NONE                                                      00000270
000028      000    C                                                                    00000280
000029      000    C       SUBROUTINES AND FUNCTION SUBPROGRAMS REQUIRED                00000290
000030      000    C          EIGEN                                                     00000300
000031      000    C                                                                    00000310
000032      000    C       METHOD                                                       00000320
000033      000    C          REFER TO W. W. COOLEY AND P. R. LOHNES, 'MULTIVARIATE PRO- 00000330
000034      000    C          CEDURES FOR THE BEHAVIORAL SCIENCES', JOHN WILEY AND SONS, 00000340
000035      000    C          1962, CHAPTER 3.                                          00000350
000036      000    C                                                                    00000360
000037      000    C    ................................................................00000370
000038      000    C                                                                    00000380
000039      000    C                                                                    00000400
000040      000    C    ................................................................00000410
000041      000    C                                                                    00000420
000042      000    C       IF A DOUBLE PRECISION VERSION OF THIS ROUTINE IS DESIRED, THE 00000430
000043      000    C       C IN COLUMN 1 SHOULD BE REMOVED FROM THE DOUBLE PRECISION    00000440
000044      000    C       STATEMENT WHICH FOLLOWS.                                     00000450
000045      000    C                                                                    00000460
000046      002         DOUBLE PRECISION A,B,XL,X,SUMV
000047      003         DIMENSION A(1),B(1),XL(1),X(1)
000048      000    C                                                                    00000480
```

```
000   C        THE C MUST ALSO BE REMOVED FROM DOUBLE PRECISION STATEMENTS    00000490
000   C        APPEARING IN OTHER ROUTINES USED IN CONJUNCTION WITH THIS      00000500
000   C        ROUTINE.                                                       00000510
000   C                                                                       00000520
000   C        THE DOUBLE PRECISION VERSION OF THIS SUBROUTINE MUST ALSO      00000530
000   C        CONTAIN DOUBLE PRECISION FORTRAN FUNCTIONS.  SQRT IN STATEMENTS00000540
000   C        110 AND 175 MUST UF CHANGED TO DSQRT.  ABS IN STATEMENT 110    00000550
000   C        MUST BE CHANGED TO DABS.                                       00000560
000   C                                                                       00000570
000   C        .......................................................... 00000580
000   C                                                                       00000590
000   C     COMPUTE EIGENVALUES AND EIGENVECTORS OF B                         00000600
000   C                                                                       00000610
000         K=1                                                               00000620
000         DO 100 J=2,M                                                      00000630
000         L=M*(J-1)                                                         00000640
000         DO 100 I=1,J                                                      00000650
000         L=L+1                                                             00000660
000         K=K+1                                                             00000670
000     100 B(K)=B(L)                                                         00000680
000   C                                                                       00000690
000   C        THE MATRIX B IS A REAL SYMMETRIC MATRIX.                       00000700
000   C                                                                       00000710
000         MV=0                                                              00000720
000         CALL EIGEN (B,X,M,MV)                                             00000730
000   C                                                                       00000740
000   C     FORM RECIPROCALS OF SQUARE ROOT OF EIGENVALUES.  THE RESULTS      00000750
000   C     ARE PREMULTIPLIED BY THE ASSOCIATED EIGENVECTORS.                 00000760
000   C                                                                       00000770
000         L=0                                                               00000780
000         DO 110 J=1,M                                                      00000790
000         L=L+J                                                             00000800
000     110 XL(J)=1.0/DSQRT(DABS(B(L)))                                       00000810
000         K=0                                                               00000820
000         DO 115 J=1,M                                                      00000830
000         DO 115 I=1,M                                                      00000840
000         K=K+1                                                             00000850
000     115 B(K)=X(K)*XL(J)                                                   00000860
000   C                                                                       00000870
000   C     FORM (B**(-1/2))PRIME * A * (B**(-1/2))                           00000880
000   C                                                                       00000890
000         DO 120 I=1,M                                                      00000900
000         N2=0                                                              00000910
000         DO 120 J=1,M                                                      00000920
000         N1=M*(I-1)                                                        00000930
000         L=M*(J-1)+I                                                       00000940
000         X(L)=0.0                                                          00000950
000         DO 120 K=1,M                                                      00000960
000         N1=N1+1                                                           00000970
000         N2=N2+1                                                           00000980
```

```
  120 X(L)=X(L)+B(N1)*A(N2)
      L=0
      DO 130 J=1,M
      DO 130 I=1,J
      N1=I-M
      N2=M*(J-1)
      L=L+1
      A(L)=0.0
      DO 130 K=1,M
      N1=N1+M
      N2=N2+1
  130 A(L)=A(L)+X(N1)*B(N2)
C
C     COMPUTE EIGENVALUES AND EIGENVECTORS OF A
C
      CALL EIGEN (A,X,M,MV)
      L=0
      DO 140 I=1,M
      L=L+I
  140 X(I)=A(L)
C
C     [illegible]UTE THE NORMALIZED EIGENVECTORS
C
      DO 150 I=1,M
      N2=0
      DO 150 J=1,M
      N1=I-M
      L=M*(J-1)+I
      A(L)=0.0
      DO 150 K=1,M
      N1=N1+M
      N2=N2+1
  150 A(L)=A(L)+B(N1)*X(N2)
C
C     AT THIS POINT, THE ROUTINE DIFFERS FROM THE STANDARD IBM NROOT.
C     THE EIGENVECTORS ARE NOT NORMALIZED TO UNIT LENGTH. AS A RESULT,
C     INSTEAD OF (X-PRIME)*(X)=I, WE HAVE (X-PRIME)*B*(X)=I , SO THAT
C     THE DISCRIMINANT SCORES WILL HAVE UNIT VARIANCES WITHIN-GROUPS.
      N2 = M*M
      DO 160 I=1,N2
  160 X(I)=A(I)
      RETURN
      END

END ELT.
```

```
@ELT,L NORMTX.OUTPUT/CLUSTER
ELT 8R1 S7401C 11/04/80 13:23:17 (25)
    1.         21           SUBROUTINE OUTPUT (LAMBDA,MEAN,COVIN,R,S,W,T,X)
    2.         21           IMPLICIT REAL*8 (A-H,O-Z)                                              00000030
    3.         21           REAL*8 LAMBDA,MEAN,MU,LIKLY
    4.         21           LOGICAL NORMAP                                                         00000060
    5.         21           COMMON/A/LIKLY,XLIMIT,NVBLES,NSAMPL,NTYPES,NSEMI,NPARAM,NORMAP,KX 00000070
    6.         21           COMMON/B/CONVRG,CRITRN,RATIO,HTEST,OLDLIK,EN,BIGEST,THRESH,GUARD, 00000080
    7.         21          1ENMIN,TITLE(18),FMT( 8),    MAXTYP,MAXAV,NCLUST,KSCRAT,INP,KPUN,KV,00000090
    8.         21          2 LOT,LACK,LUCK,IPRINT,LTIME,MAXT,ITER,ITERX,LTYE,ITMC,NRANK,IVALUF00000100
    9.         21          3,METHOD,NTYP(20)                                                      00000110
   10.         21           COMMON/F/IMAP,IUNITS,IPROF,IBACK,IPOL(20),SCAL(20),POWER,IWIND(2),
   11.         21          ,NWIND
   12.         25           COMMON/H/SBACK(20,20)
   13.         21           DIMENSION MEAN(NVBLES,NTYPES),COVIN(NSEMI,NTYPES),LAMBDA(1),      00000120
   14.         21          1 R(1),S(1),W(1),T(NVBLES,NVBLES),ALPHA(20),BETA(20),
   15.         21          ,X(NVBLES,NSAMPL)
   16.         21           CALL TYTLE                                                             00000140
   17.         21           NTP=NTYPES-1                                                           00000150
   18.         21           NOR=1
   19.         21           IF(NORMAP)NOR=0
   20.         21           INR=11
   21.         21           KTYPES=NTYPES                                                          00000160
   22.         21           IF(NORMAP) KTYPES=2                                                    00000170
   23.         21           PRINT 9, NSAMPL,NVBLES,NTP,ITER,NTP,LIKLY                              00000180
   24.         21           IF(IPROF.NE.1)GO TO 42
   25.         21    42     DO 78 L = 1,KTYPES                                                     00000190
   26.         21           IF (L-1) 1,1,2                                                         00000200
   27.         21    1      PRINT 10                                                               00000210
   28.         21           GO TO 3                                                                00000220
   29.         21    2      LTYP=L-1                                                               00000230
   30.         21           IF(NORMAP) GO TO 20                                                    00000240
   31.         21           PRINT 11, LTYP,LAMBDA(L)                                               00000250
   32.         21           GO TO 3                                                                00000260
   33.         21    20     PRINT 21                                                               00000270
   34.         21     3     LQQ=1                                                                  00000280
   35.         21           DO 4 I=1,NVBLES                                                        00000290
   36.         21           S(I)=DSQRT(COVIN(LQQ,L))                                               00000300
   37.         21    4      LQQ=LQQ+NVBLES-I+1                                                     00000310
   38.         21           IP=0
   39.         23           IWIND1=IWIND(1)
   40.         22           IF(NORMAP.AND.L.NE.1)GO TO 28
   41.         23           IF(IUNITS.EQ.1)CALL TABLE(NVBLES,IUNITS,IP,IWIND1)
   42.         21           PRINT 12, (I,I=1,NVBLES)
   43.         21           PRINT 13,(MEAN(I,L),I=1,NVBLES)                                        00000340
   44.         21     28    PRINT 14, (I,I=1,NVBLES)
   45.         21           PRINT 13, (S(I),I=1,NVBLES)                                            00000360
   46.         24    C
   47.         24    C      STORE S IN SBACK FOR VECTOR STANDARD DEVIATION BACKTRANSFORM
   48.         24    C      IN ASSIGN
   49.         24    C
   50.         24           DO 31 I=1,NVBLES
   51.         24    31     SBACK(I,L)=S(I)
   52.         21           PRINT 15, (I,I=1,NVBLES)
   53.         21           IF(IMAP.NE.1.OR.NTP.LT.2.OR.ITER.EQ.0)GO TO 30
   54.         21           IF(L.EQ.2)WRITE(INR,200)NTP,NOR
   55.         21    200    FORMAT(2I5)
```

```
  56.        21          IF(L.EQ.2)WRITE(INR,300)(LAMBDA(I+1),I=1,NTP)
  57.        21    300   FORMAT(10F5.3)
  58.        21          WRITE(INR,100)(MEAN(I,L),T=1,NVBLES)
  59.        21          WRITE(INR,100)(S(I),T=1,NVBLES)
  60.        21    100   FORMAT(6E12.5,/,6E12.5)
  61.        21    30    DO 8 I=1,NVBLES                                                   00000380
  62.        21          KJ=I-1                                                            00000390
  63.        21          IJ=I                                                              00000400
  64.        21          IF (KJ.EQ.0) GO TO 6                                              00000410
  65.        21          DO 5 J=1,KJ                                                       00000420
  66.        21          R(J)=COVIN(IJ,L)/(S(I)*S(J))                                      00000430
  67.        21    5     IJ=IJ+NVBLES-J                                                    00000440
  68.        21     6    DO 7 J = I,NVBLES                                                 00000450
  69.        21          R(J)=COVIN(IJ,L)/(S(I)*S(J))                                      00000460
  70.        21    7     IJ=IJ+1                                                           00000470
  71.        21          IF(IMAP.NE.1.OR.NTP.LT.2.OR.ITER.EQ.0.OR.L.EQ.1)GO TO 8
  72.        21          WRITE(INR,100)(R(J),J=1,NVBLES)
  73.        21    8     PRINT 13, (R(J),J=1,NVBLES)                                       00000480
  74.        21      78  CONTINUE                                                          00000490
  75.        21          IF(IBACK.LT.1.OR.ITER.LT.1)GO TO 80
  76.        21          PRINT 41,(SCAL(I),I=1,NVBLES)
  77.        21    41    FORMAT(/T10,'BACKTRANSFORMATION TERMS FOR EACH VARIATE ARE'
  78.        21         *(/T5,10E12.5))
  79.        21          IF(NORMAP)GO TO 79
  80.        21    80    IF(NTP.GT.1.AND.NTP.LE.3.AND.IPROF.EQ.1.AND.ITER.GT.0)
  81.        21         *CALL PROFIL(LAMBDA,MEAN,COVIN,NTP,X)
  82.        21    79    IF(.NOT. NORMAP) RETURN                                           00000500
  83.        21          PRINT 16, (I,I=1,NVBLES)                                          00000510
  84.        21          DO 29 L=2,NTYPES                                                  00000520
  85.        21          LTYP=L-1                                                          00000530
  86.        21    29    PRINT 17, LTYP,LAMBDA(L),(MEAN(I,L),I=1,NVBLES)                   00000540
  87.        21          RETURN                                                            00000550
  88.        21    C                                                                       00000560
  89.        21    C                                                                       00000570
  90.        21    9     FORMAT (/10X,13HSAMPLE SIZE =I11/10X,20HNUMBER OF VARIATES =I4/10X00000580
  91.        21         1,17HNUMBER OF TYPES =I7//25X,16HITERATION NUMBERI4//7X,
  92.        21         217HLOG LIKELIHOOD OFI3,23H TYPES IN THIS SAMPLE =E18.8/)            00000600
  93.        21    10    FORMAT (49X,35HCHARACTERISTICS OF THE WHOLE SAMPLE//)             00000610
  94.        21    11    FORMAT (/20X,23HCHARACTERISTICS OF TYPEI4//15X,48HTHE PROPORTION 000000620
  95.        21         1F THE POPULATION FROM THIS TYPE=F6.3/)                            00000630
  96.        21    12    FORMAT(64X,5HMEANS/I8,11I11/)
  97.        21    13    FORMAT (12F11.4)
  98.        21    14    FORMAT (/57X,19HSTANDARD DEVIATIONS/I8,11I11/)
  99.        21    15    FORMAT (/62X,12HCORRELATIONS/10(I8,11I11/))
 100.        21    21    FORMAT (//43X,49HWITHIN-GROUP STANDARD DEVIATIONS AND COVARIANCES
 101.        21         *)
 102.        21    16    FORMAT (//60X,5HMEANS/12H TYPE LAMBDA,I7,12I9/12X,I7,12I9/)
 103.        21    17    FORMAT (I4,F8.4,12F9.4/10(12X,12F9.4/))
 104.        21          END                                                               00000740

END ELT.  ERRORS: NONE. TIME:  0.833 SEC. IMAGE COUNT: 104

@FIN
```

```
@ELT,L N.PROFILE/CLUSTER
ELT007 SL73R1 02/25/80 16:09:14 (60,)
       056             SUBROUTINE PROFIL(LAMBDA,MEAN,COVIN,NTP,X)
       056             IMPLICIT REAL*8(A-H,O-Z)
       056             REAL*8 LAMBDA,MEAN,LIKLY
       056             LOGICAL NORMAP
       056             COMMON/A/LIKLY,XLIMIT,NVBLES,NSAMPL,NTYPES,NSEMI,NPARAM,NORMAP,KX
       060             COMMON/F/IMAP,IUNITS,IPROF,IBACK,IPOL(20),SCAL(20),POWER,IWIND(2),
       060            *NWIND
       056             DIMENSION LAMBDA(1),MEAN(NVBLES,NTYPES),COVIN(NSEMI,NTYPES),
       056            *BOUND(20,2),DM(20,3),DU(20,3),DL(20,3),ABOUND(20,2),
       057            *X(NVBLES,NSAMPL),XX(20,300)
       056             DO 1 I=1,NVBLES
       056             DO 1 J=1,NSAMPL
       056     1       XX(I,J)=X(I,J)
       056     C
       056     C       COMPUTE UPPER AND LOWER 3S BOUNDS IN ORIGINAL UNITS FOR EACH
       056     C       VARIATE AND STORE IN FILE WITH UNIT NUMBER 12
       056     C
       056             N=12
       056             LQQ=1
       056             DO 10 I=1,NVBLES
       056             BOUND(I,1)=MEAN(I,1)+3.*DSQRT(COVIN(LQQ,1))
       056             BOUND(I,2)=MEAN(I,1)-3.*DSQRT(COVIN(LQQ,1))
       056     10      LQQ=LQQ+NVBLES-I+1
       056     C
       056     C       BOUND=3S LIMIT OF TRANSFORMED VALUE
       056     C       ABOUND=3S LIMIT OF BACKTRANSFORMED VALUE
       056     C
       056             DO 120 I=1,NVBLES
       056             DO 11 K=1,NVBLES
       056     11      IF(I.EQ.IPOL(K))GO TO 12
       056             GO TO 115
       056     12      IF(POWER.NE.0.)GO TO 13
       056             DO 14 J=1,NSAMPL
       056     14      XX(I,J)=DEXP(XX(I,J))-SCAL(I)
       056             GO TO 15
       056     13      POWER=1./POWER
       056             DO 16 J=1,NSAMPL
       056     16      XX(I,J)=XX(I,J)**POWER-SCAL(I)
       056     15      CONTINUE
       056     C
       056     C       COMPUTE XBAR(MEAN) AND SIGMA(STANDARD DEVIATION)
       056     C       OF ORIGINAL DATA
       056     C
       056     114     XBAR=0.0
       056             DO 200 J=1,NSAMPL
       056     200     XBAR=XBAR+XX(I,J)
       056             XBAR=XBAR/FLOAT(NSAMPL)
       056             SIGMA=0.0
       056             DO 201 J=1,NSAMPL
```

```
056          XDIF=XX(I,J)-XBAR
056   201    SIGMA=SIGMA+XDIF*XDIF
056          SIGMA=SQRT(SIGMA/FLOAT(NSAMPL-1))
056          ABOUND(I,1)=XBAR+3.*SIGMA
056          ABOUND(I,2)=XBAR-3.*SIGMA
056          GO TO 119
056   115    ABOUND(I,1)=BOUND(I,1)
056          ABOUND(I,2)=BOUND(I,2)
056   119    CONTINUE
056   120    CONTINUE
056   C
056   C      COMPUTE THE RELATIVE DISTANCE FROM THE OVERALL MEAN (IN TERMS OF
056   C      THE 3S DISTANCE BETWEEN UPPER BOUND AND OVERALL MEAN, OR 10)
056   C      FOR THE MEAN OF EACH CLUSTER (DM),THE UPPER 2S BOUND (DU), AND
056   C      THE LOWER 2S BOUND (DL)
056   C
056          LQQ=1
056          DO 40 I=1,NVBLES
056          DIFF=BOUND(I,1)-MEAN(I,1)
056          DO 30 J=1,NTP
056          DM(I,J)=(MEAN(I,J+1)-MEAN(I,1))/DIFF
056          F1=MEAN(I,J+1)+2.*DSQRT(COVIN(LQQ,J+1))
056          F2=MEAN(I,J+1)-2.*DSQRT(COVIN(LQQ,J+1))
056          DU(I,J)=(F1-MEAN(I,1))/DIFF
056          DL(I,J)=(F2-MEAN(I,1))/DIFF
056   30     CONTINUE
056   40     LQQ=LQQ+NVBLES-I+1
056   37     WRITE(N,45)NVBLES,NTP
056          WRITE(N,50)(LAMBDA(I+1),I=1,NTP)
056          DO 60 J=1,2
056   60     WRITE(N,500)(ABOUND(I,J),I=1,NVBLES)
056   45     FORMAT(2I5)
056   50     FORMAT(10F6.3)
056          DO 130 I=1,NVBLES
056          WRITE(N,600)(DM(I,J),DU(I,J),DL(I,J),J=1,NTP)
059   130    CONTINUE
056   500    FORMAT(10E12.5)
056   600    FORMAT(3E20.5)
056          RETURN
056          END

END ELT.
```

```
@ELT,L N.PRTTYP/CLUSTER
ELT007 SL73R1 02/25/80 16:09:15 (10,)                                                      00000010
0CC001         6         SUBROUTINE PRTTYP (NLOT,NK,YMAX)                                   00000060
00C002         6         IMPLICIT REAL*8 (A-H,O-Z)                                          00000020
00C003         6         COMMON/B/CONVRG,CRITRN,RATIO,HTEST,OLDLIK,EN,BIGEST,THRESH,GUARD,  00000030
0CC004         6        1ENMIN,TITLE(18),FMT( 8),    MAXTYP,MAXAV,NCLUST,KSCRAT,INP,KPUN,KV,00000040
00C005         6        2 LOT,LACK,LUCK,IPRINT,LTIME,MAXT,ITER,ITERX,LTYE,ITMC,NRANK,IVALUE00000050
00C006         9        3,METHOD,NTYP(20)                                                   00000070
00C007         6         DIMENSION REC(20),NLOT(500),NK(500),YMAX(500),MSG(6)               00000080
00C008         6         DATA MSG/'TP7 ','INPU','T SE','QUEN','CE E','RROR'/
00C009         0         PRINT 301
0CC010         0 301     FORMAT(///1X,'NORMAL EXIT FROM NORMIX PROGRAM')
0CC011         0 C 300 FORMAT(1X,'IN SUBROUTINE PRTTYP')                                    00000090
00C012         6         ENDFILE 8
00C013         6         IF(INP.NE.5)REWIND INP
00C014         6         ENDFILE 11
00C015         8         ENDFILE 12                                                          00000100
00C016         6         REWIND INP                                                          00000110
0CC017         6         DO 10 I = 1,11                                                      00000120
0CC018         6         READ (INP,7) REC                                                    00000130
000019         6    10   CONTINUE                                                            00000140
00C020         6 C       CALL SLASH (REC,NBLK)                                               00000150
00C021         6         REWIND 8                                                            00000160
00C022         6         IEND = 0                                                            00000170
000023         6         LNTP = 0                                                            00000180
0CC024         6    20   IL = 0                                                              00000190
0CC025         6    30   READ (8,8,END=50) NTP,LOT,K,XMAX                                    00000200
0CC026         6         IF (NTP.EQ.0) GO TO 190                                             00000210
0CC027         6         IF (LNTP.NE.0) GO TO 40                                             00000220
0CC028         6         LNTP = NTP                                                          00000230
00C029         6    40   IF (LNTP.NE.NTP) GO TO 60                                           00000240
00C030         6         IL = IL*1                                                           00000250
00C031         6         NLOT(IL) = LOT-1                                                    00000260
00C032         6         NK(IL) = K                                                          00000270
00C033         6          YMAX(IL) = XMAX                                                    00000280
0CC034         6         GO TO 30                                                            00000290
0CC035         6    50   IEND = 1                                                            00000300
0CC036         6    60   DO 120  LT = 1,LNTP                                                 00000310
00C037         6           CALL TYTLE                                                        00000320
JC0038         6           PRINT 3, LT,LNTP                                                  00000330
00C039         6    70     REWIND INP                                                        00000340
0CC040         6           NREC=-11                                                          00000350
0CC041         6           DO 110 I = 1,IL                                                   00000360
00C042         6             IF (NLOT(I).NE.LT) GO TO110                                     00000370
00C043         6             MREC = (NK(I)-1)*NBLK+1                                         00000380
00C044         6    80       READ (INP,7,END=200) REC                                        00000390
00C045         6             NREC = NREC+1                                                   00000400
00C046         6             IF (NREC-MREC) 80,90,200                                        00000410
0CC047         6    90       PRINT 4, YMAX(I),REC                                            00000420
JC0048         6             IF (NBLK.EQ.1) GO TO 110                                        00000430
0CC049         6             DO 100 IREC = 2,NBLK
```

```
000050     6                READ (INP,7,END=200) REC                          00000440
000051     6                NREC = NREC+1                                     00000450
000052     6                PRINT 5, REC                                      00000460
000053     6   100          CONTINUE                                          00000470
000054     6   110        CONTINUE                                            00000480
000055     6   120      CONTINUE                                              00000490
000056     6          IL = 0                                                  00000500
000057     6          LNTP = NTP                                              00000510
000058     6          IF (IEND.EQ.0) GO TO 40                                 00000520
000059     6   190    REWIND 8                                                00000530
000060     6          REWIND INP                                              00000540
000061     6          RETURN                                                  00000550
000062     6   200    CALL PUTMSG (MSG,24)                                    00000560
000063     6          STOP                                                    00000570
000064     6     3    FORMAT ('  PROB.',20X,'TYPE',I2,' OF',I2,' TYPES',/)    00000580
000065     6     4    FORMAT (F8.4,3X,20A4)                                   00000590
000066     6     5    FORMAT (11X,20A4)                                       00000600
000067     6     7    FORMAT (20A4)                                           00000610
000068     7     8    FORMAT (4I4)                                            00000620
000069     6          END                                                     00000630

END ELT.
```

```
@ELT,L N.RELIST/CLUSTER
ELT007 SL73R1 02/25/80 16:09:16 (1,)
000001       000         SUBROUTINE RELIST(KPATH,NSAMPL)                                    00000010
000002       000    C                                                                       00000020
000003       000    C    GIVEN A LIST,KPATH, SUCH THAT KPATH(I) IS THE NEXT ELEMENT OF      00000030
000004       000    C    THE LIST AFTER I, THIS ROUTINE REORDERS KPATH SO THAT KPATH(I) IS  00000040
000005       000    C    THE I-TH ELEMENT OF THE LIST.                                      00000050
000006       000    C                                                                       00000060
000007       001         DIMENSION KPATH(1)                                                 00000070
000008       000         L=1                                                                00000080
000009       000         J=0                                                                00000090
000010       000    1    J=J+1                                                              00000100
000011       000         M=IABS(L)                                                          00000110
000012       000    2    IF(M-J) 3,4,5                                                      00000120
000013       000    3    M=IABS(KPATH(M))                                                   00000130
000014       000         GO TO 2                                                            00000140
000015       000    4    I=KPATH(M)                                                         00000150
000016       000         GO TO 6                                                            00000160
000017       000    5    I=KPATH(M)                                                         00000170
000018       000         KPATH(M)=KPATH(J)                                                  00000180
000019       000    6    KPATH(J)=L                                                         00000190
000020       000         L=I                                                                00000200
000021       000         IF(I.NE.0) GO TO 1                                                 00000210
000022       000         RETURN                                                             00000220
000023       000         END                                                                00000230

END ELT.
```

```
BELT,L N. REVALU/CLUSTER
ELT007 SL73R1 04/07/80 13:49:51 (29.)
000001       5      SUBROUTINE REVALU (LAMBDA,MEAN,COVIN,AV,DENSE,MEMBER,R,S)         00000010
000002       5 C    THIS SUBROUTINE RE-EVALUATES PREVIOUS ESTIMATES OF THE PARAMETERS 00000030
000003       5 C    AND FINDS INITIAL ESTIMATES PREPARATORY TO ITERATION. IT IS CALLED00000040
000004       5 C    EVERY TIME THE NUMBER OF HYPOTHESIZED TYPES IS INCREASED.         00000050
000005       5      IMPLICIT REAL*8 (A-H,O-Z)                                         00000060
000006       5      REAL*8 LAMBDA,MEAN,MU,LIKLY
000007       5      LOGICAL NORMAP                                                    00000090
000008       5      COMMON/A/LIKLY,XLIMIT,NVBLES,NSAMPL,NTYPES,NSEMI,NPARAM,NORMAP,KX 00000100
000009       5      COMMON/B/CONVRG,CRITRN,RATIO,HTEST,OLDLIK,EN,BIGEST,THRESH,GUARD, 00000110
000010       5     1ENMIN,TITLE(18),FMT( 8),    MAXTYP,MAXAV,NCLUST,KSCRAT,INP,KPUN,KV,00000120
000011       5     2 LOT,LACK,LUCK,IPRINT,LTIME,MAXT,ITER,ITERX,LTYE,ITMC,NRANK,IVALUE00000130
000012       6     3,METHOD,NTYP(20)                                                  00000140
000013       7      COMMON/G/IMN,ISTD,ICOR
000014       5      DIMENSION LAMBDA(1),MEAN(NVBLES,NTYPES),COVIN(NSEMI,NTYPES),      00000150
000015       5     1 AV(NVBLES,MAXAV),DENSE(1),MEMBER(1),R(1),S(1)                    00000160
000016       5      LTYPES=NTYPES-1                                                   00000170
000017       5      DO 1 L=2,NTYPES                                                   00000180
000018       5      LAMBDA(L)=0.0D0                                                   00000190
000019       5      DO 1 I=1,NVBLES                                                   00000200
000020       5 1    MEAN(I,L)=0.0D0                                                   00000210
000021       5      L=1                                                               00000220
000022       5      DO 2 K=1,MAXAV                                                    00000230
000023       5      IF(MEMBER(K).LT.LTYPES) L=L+1                                     00000240
000024       5      LAMBDA(L)=LAMBDA(L)+DENSE(K)                                      00000250
000025       5      DO 2 I=1,NVBLES                                                   00000260
000026       5 2    MEAN(I,L)=MEAN(I,L)+AV(I,K)*DENSE(K)                              00000270
000027       5      DO 4 L=2,NTYPES                                                   00000280
000028       5      DO 3 I=1,NVBLES                                                   00000290
000029       5 3    MEAN(I,L)=MEAN(I,L)/LAMBDA(L)                                     00000300
000030       5 4    LAMBDA(L)=LAMBDA(L)/EN                                            00000310
000031       5      IJ=0                                                              00000320
000032       5      DO 5 I=1,NVBLES                                                   00000330
000033       5      DO 5 J=I,NVBLES                                                   00000340
000034       5      IJ=IJ+1                                                           00000350
000035       5      COVIN(IJ,2)=COVIN(IJ,1)+MEAN(I,1)*MEAN(J,1)                       00000360
000036       5      DO 5 L=2,NTYPES                                                   00000370
000037       5 5    COVIN(IJ,2)=COVIN(IJ,2)-LAMBDA(L)*MEAN(I,L)*MEAN(J,L)             00000380
000038       5      IF(NORMAP) GO TO 8                                                00000390
000039       5      DO 6 L=2,NTYPES                                                   00000400
000040       5      DO 6 I=1,NSEMI                                                    00000410
000041       5 6    COVIN(I,L)=COVIN(I,2)                                             00000420
000042       5    8 IF(IVALUE.EQ.0) GO TO 18                                          00000430
000043       9      PRINT 100, IVALUE,IMN,ISTD,ICOR
000044       5   10 DO 17 L=2,NTYPES                                                  00000470
000045       5      READ (5,101,END=181)P                                             00000480
000046       5      IF(P.NE.0.0D0) LAMBDA(L)=P                                        00000490
000047       5      IF(IMN.NE.0) READ (5,101,END=182)(MEAN(I,L),I=1,NVBLES)           00000500
000048       5      IF(NORMAP) GO TO 17                                               00000510
```

```
                                                                              00000520
000049          5     11 IF(ISTD.EQ.0) GO TO 16
000050          8        READ(5,101,END=183)(S(I),I=1,NVBLFS)                 00000540
000051          5        K=0                                                  00000550
000052          5        DO 15 I=1,NVBLES                                     00000560
000053          5        IF(ICOR.NE.0) GO TO 13                               00000570
000054          5        DO 12 J=1,NVBLES                                     00000580
000055          5     12 R(J)=0.0D0                                           00000590
000056          5        R(I)=1.0D0                                           00000600
000057          5        GO TO 14
000058          8 13     READ(5,101,END=184)(R(J),J=1,NVBLES)                 00000620
000059          5     14 DO 15 J=I,NVBLES                                     00000630
000060          5        K=K+1                                                00000640
000061          5     15 COVIN(K,L)=R(J)*S(J)*S(I)                            00000650
000062          5     16 IF(NORMAP) RETURN                                    00000660
000063          5     17 CONTINUE                                             00000670
000064          5        IF(.NOT.NORMAP) RETURN                               00000680
000065          5        L=2                                                  00000690
000066          5        GO TO 11                                             00000700
000067          5     18 RFTURN
000068          7    100 FORMAT(/T10,'PARAMETERS FOR PPELIMINARY ESTIMATES ARE',5X,4I5)
                                                                              00000720
000069          5    101 FORMAT(10F8.4)
000070          5        RETURN
000071          5 181    WRITE(6,191)
000072          5        RETURN
000073          5 191    FORMAT(T2,'END OF FILE REACHFD WHEN READING P')
000074          5 182    WRITE(6,192)
000075          5        RFTURN
000076          5 192    FORMAT(T2,'END OF FILE REACHED WHEN READING MEAN(I,L)')
000077          5 183    WRITE(6,193)
000078          5        RETURN
000079          5 193    FORMAT(T2,'END OF FILE REACHED WHEN READING S')
000080          5 184    WRITE(6,194)
000081          5        RETURN
000082          5 194    FORMAT(T2,'END OF FILE REACHIT AT LABEL 13')
000083          5        RETURN                                               00000730
000084          5        END
END ELT.

@FIN
```

```
@ELT,L N.SYMINV/CLUSTER
EL1007 SL73R1 02/25/80 16:09:18 (14,)
000001        0         SUBROUTINE SYMINV (SYMTRX,NORDER,DETERM,RDUMY,SDUMY,UNTRYD,LBEFOR)00000010
000002        0         IMPLICIT REAL*8 (A-H,O-Z)                                          00000020
000003        0 C       ACM ALGORITHM 150 BY H. RUTISHAUSER                                 00000030
000004        0 C       THIS PROGRAM REPLACES A SYMMETRIC MATRIX BY ITS INVERSE.           00000040
000005        0 C       SYMTRX IS THE SYMMETRIC MATRIX, STORED IN UPPER TRIANGULAR FORM AS00000050
000006        0 C       A ONE-DIMENSIONAL ARRAY STARTING AT SYMTRX(LBEFOR+1)  .            00000060
000007        0 C       RDUMY AND SDUMY ARE TEMPORARY STORAGE VECTORS USED IN THE PROGRAM.00000070
000008        0 C       NORDER IS THE ORDER OF THE MATRIX AND THE TEMPORARY STORAGE AREAS.00000080
000009        0 C       DETERM= NATURAL LOGARITHM OF THE DETERMINANT                       00000090
000010        0         DIMENSION SYMTRX(1), RDUMY(1), SDUMY(1), UNTRYD(1)                 00000100
000011        0         DETERM=0.0                                                         00000110
000012        0         DO 1 I=1,NORDER                                                    00000120
000013        0 C       UNTRYD(I)=1 BEFORE I HAS BEEN USED AS A PIVOT, AND =0 AFTERWARDS. 00000130
000014        0 1       UNTRYD(I)=1.                                                       00000140
000015        0         DO 7 L=1,NORDER                                                    00000150
000016        0 C       SEARCH FOR PIVOT                                                   00000160
000017        0         BIGAJJ=0.0                                                         00000170
000018        0         N=LBEFOR+1                                                         00000180
000019        0         DO 2 J=1,NORDER                                                    00000190
000020        0         T=DABS(SYMTRX(N))                                                  00000200
000021        0         IF (UNTRYD(J).EQ.0..OR.T.LE.BIGAJJ) GO TO 2                        00000210
000022        0         BIGAJJ=T                                                           00000220
000023        0         K=J                                                                00000230
000024        0         NK=N                                                               00000240
000025        2 2       N=N+NORDER-J+1                                                     00000250
000026        0         IF (BIGAJJ.LE.1.0E-36) GO TO 8                                     00000260
000027        0         DETERM=DETERM+DLOG(DABS(SYMTRX(NK)))                               00000270
000028        0 C        SWEEPS OUT THE SYMMETRIC MATRIX SYMTRX FOR A SELECTED PIVOT K,    00000280
000029        0 C       RESULTING IN THE SO-CALLED PARTIAL INVERSE USEFUL IN STEPWISE      00000290
000030        0 C       REGRESSION.                                                        00000300
000031        0         UNTRYD(K)=0.                                                       00000310
000032        0 C       PREPARATION OF ELIMINATION STEP                                    00000320
000033        0         RDUMY(K)=1.0/SYMTRX(NK)                                            00000330
000034        0         SDUMY(K)=1.0                                                       00000340
000035        0         SYMTRX(NK)=0.0                                                     00000350
000036        0         N=LBEFOR+K                                                         00000360
000037        0         IF (K.EQ.1) GO TO 4                                                00000370
000038        0         KA=K-1                                                             00000380
000039        0         DO 3 J=1,KA                                                        00000390
000040        0         SDUMY(J)=SYMTRX(N)                                                 00000400
000041        0         RDUMY(J)=SYMTRX(N)*RDUMY(K)                                        00000410
000042        0         IF (UNTRYD(J).EQ.1.) RDUMY(J) = -RDUMY(J)                          00000420
000043        0         SYMTRX(N)=0.0                                                      00000430
000044        0 3       N=N+NORDER-J                                                       00000440
000045        0 4       IF (K.EQ.NORDER) GO TO 6                                           00000450
000046        0         KA=K+1                                                             00000460
000047        0         DO 5 J=KA,NORDER                                                   00000470
000048        0         N=N+1                                                              00000480
000049        0         SDUMY(J)=SYMTRX(N)                                                 00000490
```

```
0         RDUMY(J)=-SYMTRX(N)*RDUMY(K)                          00000500
0         IF (UNTRYD(J).EQ.0.) SDUMY(J) = -SDUMY(J)             00000510
0 5       SYMTRX(N)=0.0                                         00000520
0 C       ELIMINATION PROPER                                    00000530
0 6       N=LBEFOR                                              00000540
0         DO 7 I=1,NORDER                                       00000550
0         DO 7 J=I,NORDER                                       00000560
0         N=N+1                                                 00000570
0 7       SYMTRX(N)=SYMTRX(N)+SDUMY(I)*RDUMY(J)                 00000580
0   8     RETURN                                                00000590
0         END                                                   00000600

END ELT.
```

```
@ELT,L NORMTX.TABLE/CLUSTER
ELT 8R1 S74Q1C 10/23/80 09:07:24 (24)
    1.          20            SUBROUTINE TABLE(N,I,IP,II)
    2.          20            CHARACTER*10 ABBR(15),UNITS(15)
    3.          20            CHARACTER*20 NAME(15)
    4.          20       C
    5.          20       C    WHEN I EQUALS 0, THIS SUBROUTINE READS NAME, ABBREVIATED NAME,
    6.          20       C    AND UNITS FOR EACH VARIATE AND PRINTS THEM IN A TABLE.
    7.          20       C    WHEN I EQUALS 1, IT PROVIDES NAME, ABBREVIATED NAME, AND UNITS
    8.          20       C    AS HEADINGS FOR TABLE COLUMNS IN OUTPUT AND ASSIGN SUBROUTINES.
    9.          20       C    IF IP EQUALS 0, THE WIND VARIATES ARE W COMP AND S COMP.
   10.          20       C    II IS SET EQUAL TO THE FIRST WIND VARIATE NUMBER.
   11.          20       C    IF IP EQUALS 1, IT PROVIDES THE SAME INFORMATION AS WHEN
   12.          20       C    IP EQUALS 0, BUT THE WIND VARIATES BECOME SPEED AND DIRECTION,
   13.          20       C    RESPECTIVELY.
   14.          20       C
   15.          20            IF(I.EQ.1)GO TO 40
   16.          20            PRINT 10
   17.          20            DO 20 K=1,N
   18.          20            READ(5,100)J,NAME(K),ABBR(K),UNITS(K)
   19.          20       20   WRITE(6,200)J,NAME(K),ABBR(K),UNITS(K)
   20.          20            RETURN
   21.          20       40   IF(IP.EQ.1)ABBR(II)='     SPEED'
   22.          20            IF(IP.EQ.1)ABBR(II+1)='     DIREC'
   23.          20            IF(IP.EQ.1)UNITS(II+1)='       DEG'
   24.          20            IF(IP.EQ.1)GO TO 60
   25.          22            IF(IP.EQ.0.AND.II.EQ.0)GO TO 60
   26.          21            IF(IP.EQ.0)ABBR(II)='    W COMP'
   27.          21            IF(IP.EQ.0)ABBR(II+1)='    S COMP'
   28.          24            IF(IP.EQ.0)UNITS(II+1)= UNITS(II)
   29.          20       60   WRITE(6,300)(ABBR(J),J=1,N)
   30.          20            WRITE(6,300)(UNITS(J),J=1,N)
   31.          20            RETURN
   32.          20       10   FORMAT(//45X,'VARIATE NAMES AND UNITS'/)
   33.          20       100  FORMAT(I10,A20,2A10)
   34.          20       200  FORMAT(27X,I10,A20,2A10)
   35.          20       300  FORMAT(12(1X,A10))
   36.          20            END

 END ELT.  ERRORS: NONE. TIME:  0.872 SEC. IMAGE COUNT: 36

 @FIN
```

```
@ELT,L N.TYTLE/CLUSTER
EL1007 SL73R1 02/25/80 16:09:20 (13,)
COC001       009           SUBROUTINE TYTLE                                                    00000010
00C002       009           IMPLICIT REAL*8 (A-H,O-Z)                                           00000020
00C003       009           REAL*3 LAMBDA,MEAN,MU,LIKLY
00C004       009           LOGICAL NORMAP                                                      00000040
00C005       009           COMMON/A/LIKLY,XLIMIT,NVBLES,NSAMPL,NTYPES,NSEMI,NPARAM,NORMAP,KX   00000050
0CC006       009           COMMON/B/CONVRG,CRITRN,RATIO,HTEST,OLDLIK,EN,BIGEST,THRESH,GUARD,   00000060
0CC007       009          1ENMIN,TITLE(18),FMT( 8),    MAXTYP,MAXAV,NCLUST,KSCRAT,INP,KPUN,KV,00000070
0CC008       009          2 LOT,LACK,LUCK,IPRINT,LTIME,MAXT,ITER,ITERX,LTYE,ITMC,NRANK,IVALUE00000080
00C009       013          3,METHOD,NTYP(20)                                                    00000090
0CC010       009           PRINT 1                                                             00000100
00C011       009           IF(NORMAP) PRINT 2                                                  00000110
00C012       009           PRINT 3,TITLE                                                       00000120
00C013       009           RETURN                                                              00000130
00C014       009    C                                                                          00000140
00C015       009    C                                                                          00000150
00C016       011        1  FORMAT (1H1,52X,14HPROGRAM NORMIX/33X,54HWOLFE NORMAL MIXTURE ANAL00000160
0CC017       012          1YSIS PROCEDURE(1979 REVISION))                                      00000170
0CC018       011    2      FORMAT(40X,39HEQUAL COVARIANCE MATRICES WITHIN GROUPS)              00000180
0CC019       010    3      FORMAT(/(1X,9A8))
0CC020       009           END                                                                 00000200

END ELT.
```

```
@ELT,L N.UPLOW/CLUSTER
EL1007 SL73R1 02/25/80 16:09:21 (0,)
00C001      000         SUBROUTINE UPLOW (A,T,N)                                           00000010
00C002      000   C  THIS VERSION IS A REWRITE BY JOINER AUG 29/73                         00000020
00C003      000   C  CHANGES A SYMETRIC MATRIX FROM UPPER TO LOWER TRIANGULAR FORM         00000030
00C004      000   C  ASSUMING THE MATRIX CONTAINS VARIANCE-COVARIANCE DATA                 00000040
00C005      000   C  IT IS CHANGED AS FOLLOWS                                              00000050
00C006      000   C           AT CALL TO SUBROUTINE                 UPON RETURN            00000060
00C007      000   C     A(1)      A(2)      A(3)      A(4)      A(1)                       00000070
00C008      000   C     1VS1      1VS2      1VS3      1VS4      1VS1                       00000080
0CC009      000   C               A(5)      A(6)      A(7)      A(2)      A(3)             00000090
00C010      000   C               2VS2      2VS3      2VS4      1VS2      2VS2             00000100
0CC011      000   C                         A(8)      A(9)      A(4)      A(5)      A(6)   00000110
00C012      000   C                         3VS3      3VS4      1VS3      2VS3      3VS3   00000120
00C013      000   C                                   A(10)     A(7)      A(8)      A(9)   00000130
0CC014      000   C                                   4VS4      1VS4      2VS4      3VS4   00000140
00C015      000   C                                                                        00000150
0C0016      000   C                                                                        00000160
00C017      000         REAL*8 A,T                                                         00000170
00C018      000         DIMENSION A(1),T(1)                                                00000180
0CC019      000         IF (N.LT.3) GO TO 90                                               00000190
00C020      000         IR = 0                                                             00000200
00C021      000         IS = 0                                                             00000210
00C022      000         I1 = 0                                                             00000220
00C023      000         DO 10 I = 1,N                                                      00000230
00C024      000         IR = I1 + I                                                        00000240
000025      000         I1 = IR                                                            00000250
00C026      000         I2 = I                                                             00000260
00C027      000         DO 10 J = 1,N                                                      00000270
00C028      000         IS = IS+1                                                          00000280
0CC029      000         T(IR) = A(IS)                                                      00000290
00C030      000         IR = IR + I2                                                       00000300
00C031      000      10 I2 = I2 + 1                                                        00000310
00C032      000         NN = N*(N+1)/2                                                     00000320
000033      000         DO 20 I = 1,NN                                                     00000330
00C034      000      20 A(I) = T(I)                                                        00000340
00C035      000      90 RETURN                                                             00000350
00C036      000         END                                                                00000360

END ELT.
```

```
@ELT,L N.PUTMSG/CLUSTER
ELT007 SL73R1 02/25/80 16:09:22 (2,)
00C001       2       SUBROUTINE PUTMSG(MSG,NCHAR)
000002       0       DIMENSION MSG(1)
00C003       0 C---PRINT NCHAR FROM ARRAY MSG.
00C004       0       NWORDS=NCHAR/4+1
0CC005       0       PRINT 1001,(MSG(I),I=1,NWORDS)
00C006       1 1001  FORMAT(18A4)
00C007       0       END

END ELT.
```

```
@ELT,L N.NORM
EL1007 SL73R1 02/25/80 16:09:23 (12,)
000001        008          SUBROUTINE NORM(YY,P)
000002        011          COMMON/C/YBAR,SIGMA
000003        008          YY=(YY-YBAR)/SIGMA
000004        008          P=.5*ERFC(-YY*.7071068)
000005        008          RETURN
000006        008          END

END ELT.

@FIN
```

6.2 APPENDIX B. LISTING OF NORMIX-PREP SUBROUTINES

Preceding page blank

```
@ELT,L NORMIX-PREP.MAIN
ELT007 SL73R1 02/29/80 09:03:15 (43,)
039           REAL*4 MIN,MINPOL
039           DOUBLE PRECISION DSEED
039           COMMON/CMEAN/YBAR,SIGMA
039           DIMENSION IPOL(20),IWIND(2),Y(500,20),CELLS(20),COMP(20),X(500),
039          1          WORK1(110),WORK2(27),IWORK(50),MINMOL(20),THRESH(20),
039          2          MIN(20),TITLE(36),IQOL(20),BACK(20)
039           CHARACTER*60 FORMTI,FORMTO
039           NAMELIST/LIST/NVBLES,NSAMPL,NPOL,IPOL,NWIND,IWIND,FORMTI,FORMTO,
039          1                IHIST,IRAND,IQOL,DEG,NCELLS,IDF,ISPACE,MINPOL,
039          2                THRESH,NCLASS,IUNIT,IPROB,FACTOR,POWER
039     C
039     C     EXTERNAL FUNCTION NORM IS REQUIRED FOR IMSL SUBROUTINE GFIT
039     C
039           EXTERNAL NORM
039           READ(5,6,END=7)TITLE
039     6     FORMAT(18A4)
039           PRINT 1
039     1     FORMAT(1H1,50X,'PROGRAM NORMIX-PREP',/,31X,'OPTIONAL NATURAL LOG
039          *TRANSFORMATION, RANDOM NUMBER ADDITION,',/,26X,'CHI-SQUARE TEST FO
040          *R NORMALITY, EQUAL-PROBABILITY INTERVAL HISTOGRAMS,',/,24X,
043          *'PROBABILITY CHARTS, SCALING, DECOMPOSITION OF WIND SPEED INTO COM
043          *PONENTS,',/,32X,'ROTATION OF WIND AXES, AND VALUES FOR BACKTRAN
043          *TRANSFORMATION',/,30X,'OF TRANSFORMED DATA, ALL OPTIONS AVAILABLE
043          *FOR ANY VARIABLE')
039           PRINT 2,TITLE
039     2     FORMAT(///1X,(18A4))
039     7     READ(5,LIST)
039           IF(FACTOR.EQ.0.)FACTOR=1.D-5
039           WRITE(6,40)
039        40 FORMAT(1H0,T60,'INPUT PARAMETERS')
039           WRITE(6,LIST)
039           IDFSAV=IDF
039           IPASS=1
039     C
039     C     READ IN THE DATA WITH FORMAT SUPPLIED IN NAMELIST
039           DO 50 J=1,NSAMPL
039           READ(5,FORMTI) (Y(J,I),I=1,NVBLES)
039        50 CONTINUE
039      1000 IF(IPASS.EQ.1) WRITE(6,10)
039        10 FORMAT('1',T60,'INPUT VARIABLES')
039           IF(IPASS.EQ.2) WRITE(6,15)
039        15 FORMAT('1',T60,'TRANSFORMED VARIABLES')
039           WRITE(6,20)
039        20 FORMAT(//,' SAMPLE',T15,'1',T27,'2',T39,'3',T51,'4',T63,'5',
039          1         T75,'6',T87,'7',T99,'8',T111,'9',T123,'10',/)
039     C
039     C     PRINT THE DATA
039     C
039           DO 25 I=1,NSAMPL
```

```
039          WRITE(6,30) I,(Y(I,J),J=1,NVBLES)
039       30 FORMAT(1X,T2,I3,T9,10(1PG12.5),/,T9,10(1PG12.5))
039       25 CONTINUE
039   C
039   C      BEGIN LOOPING THROUGH ALL VARIABLES
039   C
039          DO 100 I=1,NVBLES
039          IF(IPASS.EQ.1) GO TO 90
039          DO 60 J=1,NVBLES
039   C
039   C      CHECK TO SEE IF VARIABLE 'I' IS TO BE TRANSFORMED.
039   C
039          IF(I.EQ.IPOL(J)) GO TO 90
039       60 CONTINUE
039          GO TO 100
039   C
039   C      COMPUTE 'YBAR' (MEAN) AND 'SIGMA' (STANDARD DEVIATION)
039   C
039       90 YBAR=0.0
039          DO 200 J=1,NSAMPL
039          YBAR=YBAR+Y(J,I)
039      200 CONTINUE
039          YBAR=YBAR/FLOAT(NSAMPL)
039          SIGMA=0.0
039          DO 300 J=1,NSAMPL
039          YDIF=Y(J,I)-YBAR
039          SIGMA=SIGMA+YDIF*YDIF
039      300 CONTINUE
039          SIGMA=SQRT(SIGMA/FLOAT(NSAMPL-1))
039   C
039   C      IF A PSEUDO-RANDOM NORMAL DEVIATE IS TO BE APPLIED, OBTAIN 'DSEED' (SEED)
039   C      AND MULTIPLY EACH RANDOM NUMBER BY 'SIGMA' AND BY 'FACTOR'.
039   C      IQOL IS A VARIABLE SPECIFIER USED WHEN IRAND=2.
039   C      WHEN IRAND=0 (DEFAULT), NO RANDOM DEVIATES ARE
039   C      APPLIED.  WHEN IRAND=1, ALL VARIABLES ARE ALTERED BY
039   C      RANDOM DEVIATES.
039   C
039          IF(IRAND.EQ.1)GO TO 390
039          IF(IRAND.EQ.0) GO TO 400
039          DO 65 J=1,NVBLES
039          IF(I.EQ.IQOL(J))GO TO 390
039   65     CONTINUE
039          GO TO 400
039   390    CALL SEED(DSEED)
039          IF(I.EQ.1)PRINT 70, DSEED
039   70     FORMAT(T10,'THE ORIGINAL RANDOM NUMBER SEED = ',F20.2)
039          IT=1
039          DO 500 J=1,NSAMPL
039          YRAND=GGNOF(DSEED)
039          Y(J,I)=Y(J,I)+FACTOR*YRAND*SIGMA
039      500 CONTINUE
```

```
039      400 IDF=IDFSAV
039          WRITE(6,450) I
039      450 FORMAT(1H1,T55,'ANALYSIS OF VARIABLE',I3)
039          WRITE(6,460) YBAR,SIGMA
039      460 FORMAT(/,T2,'MEAN VALUE',T53,'= ',F8.4,
039         1        /,T2,'STANDARD DEVIATION',T53,'= ',F8.4,/)
039     C
039     C        CALL IMSL SUBROUTINE 'GFIT' WITH ARGUMENTS 'NORM' (EXTERNAL
039     C        FUNCTION), 'NCELLS' (NUMBER OF CELLS SUPPLIED  IN NAMELIST),
039     C        'Y' (VARIABLE ARRAY), 'NSAMPL' (NUMBER OF OBSERVATIONS), 'CELLS'
039     C        (OUTPUT FREQUENCY IN EACH CELL), 'COMP' (OUTPUT COMPONENTS OF
039     C        CHI-SQUARED STATISTIC), 'CS' (OUTPUT CHI-SQUARED STATISTIC),
039     C        'IDF' (INPUT NUMBER OF HYPOTHESIZED DISTRIBUTION FUNCTION
039     C        PARAMETERS ESTIMATED FROM DATA, OUTPUT DEGREES OF FREEDOM OF
039     C        CHI-SQUARED STATISTIC CS), 'Q' (OUTPUT PROBABILITY OF CHI-SQUARED
039     C        STATISTIC EXCEEDING CHI-SQUARE IF NULL HYPOTHESIS IS TRUE),
039     C        'IER' (OUTPUT ERROR PARAMETER)
039     C
039          CALL GFIT(NORM,NCELLS,Y(1,I),NSAMPL,CELLS,COMP,CS,IDF,Q,IER)
039          WRITE(6,550)
039      550 FORMAT(//,' CELL NUMBER',5X,'FREQUENCY')
039          DO 575 J=1,NCELLS
039          WRITE(6,580) J,CELLS(J)
039      580 FORMAT(6X,I2,11X,F5.0)
039      575 CONTINUE
039          WRITE(6,590) CS,Q,IDF
039      590 FORMAT(/,T2,'CHI-SQUARE',T53,'= ',F8.4,
039         1        /,T2,'PROB. OF EXCEEDING CHI-SQUARE IF NULL HYP. IS TRUE',
039         2        T53,'= ',F8.4,/,T2,'DEGREES OF FREEDOM',T53,'= ',I3)
039          IF(IHIST.NE.1) GO TO 600
039     C
039     C        CALL IMSL SUBROUTINE 'USHIST' WITH ARGUMENTS 'CELL' (FREQUENCIES
039     C        FROM GFIT), 'NCELLS' (NUMBER OF CELLS), 'ISPACE (SPACING BETWEEN
039     C        HISTOGRAM BARS (=1,2, OR 4)),'WORK1', 'WORK2', AND 'IWORK'
039     C        (INTERNAL WORK VECTORS), 'IER' (OUTPUT ERROR PARAMETER) TO PRINT
039     C        HISTOGRAMS
039     C
039          CALL USHIST(CELLS,NCELLS,ISPACE,WORK1,WORK2,IWORK,IER)
039      600 IF(IPROB.NE.1) GO TO 100
039     C
039     C        ORDER OBSERVATIONS FOR PROBABILITY CHART
039     C
039          DO 625 J=1,NSAMPL
039          X(J)=Y(J,I)
039      625 CONTINUE
039          L=NSAMPL-1
039          DO 650 J=1,L
039          LL=L-J+1
039          DO 700 II=1,LL
039          LG=II+1
039          IF(X(II).LT.X(LG)) GO TO 700
```

```
039          AA=X(II)
039          X(II)=X(L6)
039          X(L6)=AA
039      700 CONTINUE
039      650 CONTINUE
039   C
039   C      CALL SUBROUTINE 'CUMPLT' WITH ARGUMENTS 'X' (VECTOR VARIABLE),
039   C      'NSAMPL' (NUMBER OF OBSERVATIONS), AND 'NCLASS' (NUMBER OF
039   C      CLASS INTERVALS)
039   C
039          CALL CUMPLT(X,NSAMPL,NCLASS)
039      100 CONTINUE
039          IF(IPASS.EQ.2) GO TO 2000
039   C
039   C      NPOL=SWITCH(=1 FOR SCALING)
039   C
039          IF(NPOL.EQ.0) GO TO 910
039   C
039   C      IF VARIABLE IS TO BE SCALED, FIND 'MIN' (MINIMUM VALUE)
039   C
039          DO 750 I=1,NVBLES
039          MIN(I)=1.0E+06
039      750 CONTINUE
039          DO 800 I=1,NVBLES
039          YINC=0.
039          IF(I.NE.IPOL(I))GO TO 890
039          J=IPOL(I)
039          DO 850 K=1,NSAMPL
039          IF(Y(K,J).LT.MIN(I)) MIN(I)=Y(K,J)
039      850 CONTINUE
039          IF(MIN(I).GT.THRESH(I)) GO TO 880
039   C
039   C      IF MINIMUM VALUE < 'THRESH' (THRESHOLD VALUE), ADD 'YINC' TO EACH
039   C      OBSERVATION OF VARIABLE 'H'.  'THRESH' AND 'MINPOL' (DESIRED
039   C      MINIMUM VALUE) ARE SUPPLIED IN NAMELIST
039   C
039          YINC=MINPOL(I)-MIN(I)
039          DO 875 K=1,NSAMPL
039          Y(K,J)=Y(K,J)+YINC
039      875 CONTINUE
039      880 DO 890 K=1,NSAMPL
039   C
039   C      TAKE NATURAL LOGS OR TAKE POWERS
039   C
039          IF(POWER.NE.0.)GO TO 810
039          Y(K,J)=ALOG(Y(K,J))
039          GO TO 890
039   810    Y(K,J)=Y(K,J)**POWER
039      890 CONTINUE
039   C
039   C      PRINT THE BACKTRANSFORMATION TERM, EVEN IF IT IS 0
```

```
039   C
039           BACK(I)=VINC
039     800 CONTINUE
039           PRINT 870,(BACK(I),I=1,NVBLES)
039   870     FORMAT(/T10,'BACKTRANSFORMATION TERMS',/,(T10,6E12.5))
039   C
039   C       IF 'NWIND'=1, COMPUTE 'Y(I,ISPEED)' (WEST WIND COMPONENT) AND
039   C       'Y(I,IDIR)' (SOUTH WIND COMPONENT) FOR 'IWIND(1)' (WIND SPEED
039   C       VARIABLE) AND 'IWIND(2) (WIND DIRECTION VARIABLE)
039   C
039     910 IF(NWIND.NE.1) GO TO 950
039           ISPEED=IWIND(1)
039           IDIR=IWIND(2)
039           DTRAD=0.0174533
039           DO 900 I=1,NSAMPL
039           RMAG=Y(I,ISPEED)
039   C
039   C       IF A CLOCKWISE ROTATION OF AXES IS DESIRED, SUBTRACT 'DEG'
039   C       (THE NUMBER OF DEGREES OF ROTATION) FROM THE WIND DIRECTION
039   C
039           RADIAN=(Y(I,IDIR)-DEG)*DTRAD
039           Y(I,ISPEED)=RMAG*(-SIN(RADIAN))
039           Y(I,IDIR)=RMAG*(-COS(RADIAN))
039     900 CONTINUE
039     950 IF(IPASS.EQ.2) GO TO 2000
039   C
039   C       SCALING, TRANSFORMATION, AND WIND COMPONENTS ARE HANDLED IN
039   C       THE FIRST PASS
039   C
039           IRAND=0
039           IPASS=2
039           GO TO 1000
039    2000 DO 2100 I=1,NSAMPL
039   C
039   C       STORE TRANSFORMED DATA IN FILE WITH UNIT NUMBER 'IUNIT'
039   C
039           WRITE(IUNIT,FORMTO) (Y(I,J),J=1,NVBLES)
039    2100 CONTINUE
039   C
039   C       STORE BACKTRANSFORMATION SCALING TERMS
039   C
039           WRITE(IUNIT,FORMTO)(BACK(I),I=1,NVBLES)
039   C
039   C       THIS PROGRAM WAS WRITTEN BY J. A. STEPHENSON IN OCTOBER, 1979
039   C       AND ALTERED BY M. E. GRAVES IN JANUARY, 1980
039   C
039           STOP
039           END

END ELT.

@FIN
```

```
******  CUMPLT  ******

@ELT,L NORMIX-PREP.CUMPLT
ELT007 SL73R1 02/20/80 09:11:55 (4,)
         000          SUBROUTINE CUMPLT(Y,N,NCELL)
         000     C
         000     C   THIS SUBROUTINE CAUSES CUMULATIVE PLOTS TO BE PRINTED.
         000     C
         000          DIMENSION GROUP(15),ROINT(15),NFREQ(15),Y(200),
         004         *CUM(15),POINT(16)
         004          DIMENSION CELL(1000),BLNK(113),PRNT(113),P(110)
         000          DATA BLNK/1H1,55*1H ,1H.,55*1H ,1HI/
         000          DATA P/.01,.013,.017,.02,.03,.04,.05,.06,.08,.1,.12,.14,.2,.23,
         000         *.3,.34,.4,.5,.6,.8,.9,1.,1.2,1.6,1.8,2.,2.5,3.,3.4,4.,4.6,5.,
         000         *6.,7.,8.,9.,10.,11.,13.,14.,16.,18.,19.,20.,22.,26.,28.,30.,32.,
         000         *34.,38.,40.,42.,44.,48.,50.,52.,56.,58.,60.,62.,66.,68.,70.,72.,
         000         *74.,78.,80.,81.,82.,84.,86.,87.,89.,90.,91.,92.,93.,94.,95.,
         000         *95.4,96.6,97.,97.5,98.,98.2,98.4,98.8,99.,99.1,99.2,99.4,99.5,
         000         *99.6,99.66,99.7,99.77,99.8,99.86,99.88,99.9,99.92,99.94,99.95,
         000         *99.96,99.97,99.98,99.983,99.987,99.99/
         000          DATA IIA/1HX/
         000          XY=Y(N)
         000          KD=0
         000          IF (NCELL.EQ.0) NCELL=15
         000          DO 130 I=1,NCELL
         000     130  NFREQ(I)=0
         000          H=Y(N)
         000          B=Y(1)
         000     C    SUBROUTINE SCALE
         000          R=H-B
         000     C    ERROR CHECK
         000          IFAULT=0
         000          IF (R.EQ.0.)IFAULT=1
         000          IF (NCELL.LT.1)IFAULT=2
         000          IF (IFAULT.NE.0) GO TO 5
         000     C    ESTABLISH MINIMUM VALUE
         000          KOUNT=0
         000          KOD=0
         000     21   IF (R.GT.1.) GO TO 20
         000          KOUNT=KOUNT+1
         000          R=R*10.
         000          GO TO 21
         000     20   IF (R.LE.10.) GO TO 22
         000          KOUNT=KOUNT-1
         000          R=R/10.
         000          GO TO 20
         000     22   SB=B*10.**KOUNT
         000          IF (SB.LT.0.) SB=SB-1.
         000          IB=SB
         000          SB=IB
```

```
000          SB=SB/10.**KOUNT
000   C      R IS THE NEW RANGE USING THE SCALED VALUES
000          R=H-SB
000   C      AIL IS THE INTERVAL LENGTH TO BE SCALED.
000          AIL=R/NCELL
000          KOUNT=0
000   26     IF (AIL.GT.1.) GO TO 25
000          KOUNT=KOUNT+1
000          AIL=AIL*10.
000          GO TO 26
000   25     IF (AIL.LE.10.) GO TO 27
000          KOUNT=KOUNT-1
000          AIL=AIL/10.
000          GO TO 25
000   27     IA=AIL
000   C      CHECK FOR A LESS THAN 1.5
000          AA=0.
000          IF (AIL.LT.1.5) AA=-.5
000          AIL=IA
000          AIL=AA+1.+AIL
000          AIL=AIL/10.**KOUNT
000          IF (KOD.EQ.1) GO TO 41
000   C      CHECK FOR NECESSITY OF A SECOND PASS
000          TEST=NCELL*AIL
000          TEST1=(H-B)/TEST
000          IF (TEST1.GT..8) GO TO 41
000          KOUNT=1
000          KOD=1
000          R=H-B
000          GO TO 21
000   C      FINAL OUTPUT
000   41     POINT(1)=SB
000          K=NCELL+1
000          DO 30 J=2,K
000          POINT(J)=POINT(J-1)+AIL
000       30 CONTINUE
000   C   ** ***********************************
000          RAP=POINT(K)-SB
000   5      CONTINUE
000          CELL2=NCELL*1.0
000          WIDTH=RAP/CELL2
000          RIDPT=WIDTH/2.0
000          GROUP(1)=SB+RIDPT
000          DO 160 I=2,NCELL
000   160    GROUP(I)=GROUP(I-1)+WIDTH
000          DO 190 I=1,N
000          DO 170 M=1,NCELL
000          IF (Y(I).LE.GROUP(M)) GO TO 180
000   170    CONTINUE
000   180    NFREQ(M)=NFREQ(M)+1
000   190    CONTINUE
```

```
0C0098     000          DO 191 I=1,NCELL
000099     000   191    ROINT(I)=POINT(I)
000100     000          XSUM=0.0
000101     C00          DO 220 I=1,N
000102     000   220    XSUM=XSUM+Y(I)
000103     000          AVE=XSUM/(N*1.)
0CC104     C00          PART1=0.0
000105     000          DO 230 I=1,N
000106     000   230    PART1=PART1+(Y(I)-AVE)**2
000107     000          VAR=PART1/(N*1.)
0CC108     000          VR=PART1/((N-1)*1.)
0C0109     000          SD=SQRT(VAR)
000110     000          SD1=SQRT(VR)
000111     000          XF=0.
00C112     C00          SUM=0.
000113     000          DO 299 I=1,NCELL
0C0114     000          XF=NFREQ(I)+XF
0C0115     000          CUM(I)=(XF-.5)/N*100.
0C0116     000   299    IF(XF.EQ.0.) CUM(I)=0.
000117     000   C      SUBROUTINE CUMPLT
000118     000          DO 19 I=1,N
00C119     000          CELL(I)=(Y(N+1-I)-AVE)/SD1
000120     000   19      CONTINUE
00C121     000   C      PRODUCE PROBABILITY CHART
000122     000          WRITE (6,1000)
000123     000   1000   FORMAT(1H1,54X,'CUMULATIVE ','DISTRIBUTION CHART'/)
000124     000          WRITE (6,1010)
000125     000   1010   FORMAT(13X,'.01   .05   .20  ','.50 1.0 2.0   5.0  10.'4X,
000126     000         *'20. 30.     50.     70. 80.    ','90.  95.   98. 99. 99.5 99.8  '
0C0127     000         *,'99.95 99.99')
000128     000          WRITE (6,1020)
000129     000   1020   FORMAT(14X,'I',5X,'I',5X,'I',4X,'I',3X,'I',3X,'I',5X,'I',4X,'I',
0C0130     000         *5X,' I',3X,'I',7X,'I',7X,'I',3X,'I',6X,'I',4X,'I',5X,'I',3X,'I',
000131     000         *3X,'I',4X,'I',5X,'I',5X,'I')
000132     000          WRITE (6,1030)
000133     000   103C   FORMAT(13X,1HI,111(1H-),1HI)
000134     000          DO 10 I=1,NCELL
0C0135     000          POINT(I)=(POINT(I)-AVE)/SD1
0C0136     000   10     CONTINUE
000137     000          DO 110 MO=1,NCELL
000138     000          M=NCELL-MO+1
000139     000          DO 18 K=1,113
000140     000   18     PRNT(K)=BLNK(K)
000141     000          IF (CUM(M).GE.99.99) PRNT(113)=1H+
000142     000          IF (CUM(M).GE.99.99.OR.CUM(M).EQ.0.) GO TO 60
000143     000          DO 50 L=1,110
000144     000          IF (CUM(M).GT.P(L)) GO TO 50
000145     000          PRNT(L+1)=1H+
000146     000          GO TO 60
000147     000   50     CONTINUE
0C0148     000   C      SET UP ORDINATE SCALE VALUES
```

```
000   60     CONTINUE
000          IF (ABS(XY)-1.)111,111,112
000   112    IF(ABS(XY)-10.)113,113,114
000   114    IF (ABS(XY)-100.)115,115,116
000   111    WRITE (6,1051) GROUP(M),PRNT
000          GO TO 109
000   113    WRITE (6,1052) GROUP(M),PRNT
000          GO TO 109
000   115    WRITE (6, 1053)GROUP(M),PRNT
000          GO TO 109
000   116    WRITE (6,1054) GROUP(M),PRNT
003   1051   FORMAT(1H ,4X,F7.3,1X,113A1)
000   1052   FORMAT(1H ,4X,F7.2,1X,113A1)
000   1053   FORMAT(1H ,4X,F7.1,1X,113A1)
000   1054   FORMAT(1H ,4X,F7.0,1X,113A1)
000   1070   FORMAT(1X)
000   109    CONTINUE
000          DO 16 K=1,113
000   16     PRNT(K)=BLNK(K)
000          IF(MO-7)71,73,71
000   71     IF (MO-9)72,74,72
000   73     WRITE (6,1040) (PRNT(L),L=1,113)
000          GO TO 110
000   74     WRITE (6,1042) (PRNT(L),L=1,113)
000          GO TO 110
000   72     WRITE (6,1043) (PRNT(L),L=1,113)
000   1040   FORMAT(7H LINEAR,6X,113A1)
000   1042   FORMAT(7H SCALE ,6X,113A1)
000   1043   FORMAT(13X,113A1)
000   110    CONTINUE
000          WRITE (6,1030)
000          WRITE (6,1020)
000          WRITE (6,1010)
000          WRITE (6,1080)
000   1080   FORMAT(1H0,40X,'CUMULATIVE FREQUENCY (PERCENT) - NORMAL PROBABILIT
000         *Y SCALE')
000   1      RETURN
000          END

END ELT.

@HDG,P ****** MAIN ******
```

```
@ELT,L NORMIX-PREP.NORM
ELT007 SL73R1 02/20/80 09:12:04 (10,)
000001         006           SUBROUTINE NORM(YY,P)
000002         010           COMMON/CMEAN/YBAR,SIGMA
000003         006           YY=(YY-YBAR)/SIGMA
000004         006           P=.5*ERFC(-YY*.7071068)
000005         006           RETURN
000006         006           END

END ELT.

@HDG,P  ******  SEED  ******
```

```
 ******  SEED  ******

@ELT,L NORMIX-PREP.SEED
ELT007 SL73R1 02/20/80 09:12:06 (9,)
000001       006          SUBROUTINE SEED(DSEED)
000002       006          DOUBLE PRECISION DSEED
000003       008          CALL SEED1(ITIME)
000004       008          ITIME=IABS(ITIME)
000005       009       100 IF(ITIME.LT.2147483647) GO TO 200
000006       009          ITIME=ITIME/10
000007       009          GO TO 100
000008       009       200 DSEED=DFLOAT(ITIME)
000009       005          RETURN
000010       005          END

END ELT.

@FDG,P  ******  SEED1  ******
```

```
******  SEED1  ******

@ELT,L NORMIX-PREP.SEED1
ELT007 SL73R1 02/20/80 09:12:07 (5,)
000001       001   $(0)      AXR$
000002       005   $(1),SEED1*
000003       001             SA        A0,X1
000004       001             LXM       A0,0,X1
000005       002             LXI,U     A0,0
000006       002             LXI,U     A1,1
000007       004             LXM,U     A1,14
000008       001             ER        PCT$
000009       001             J         0,X11
000010       001             END

END ELT.

@HDG,N

@FIN
```

6.3 APPENDIX C. LISTING OF PROFILE-PLOTTING ROUTINE

Preceding page blank

```
****** MAIN ******

@ELT,L PROFILE.MAIN
EL1007 SL73R1 02/21/80 15:59:43 (30,)
000001       026   C   THIS PLOTTING ROUTINE PRODUCES PROFILE PLOTS OF UP TO 20 VARIABLES
000002       026   C          AND THREE CLUSTERS
000003       026           DIMENSION ABOUND (20,2),P(3),DM(20,3),DU(20,3),DL(20,3)
000004       026           CHARACTER * 12 TITLE/'PROFILE PLOT'/
000005       026           CHARACTER * 4 PLUS/'X+3S'/
000006       026           CHARACTER *6 MEAN/'X OR X'/
000007       026           CHARACTER * 4 MINUS/'X-3S'/
000008       026           CHARACTER * 1 BAR/'-'/
000009       026           CHARACTER * 6 BARS/'-     -'/
000010       026           CHARACTER * 1 G/'G'/
000011       026           CHARACTER * 2 VARLAB(21)/' 1',' 2',' 3',' 4',' 5',' 6',' 7',' 8',
000012       026          *' 9','10','11','12','13','14','15','16','17','18','19','20','21'/
000013       026           REAL COUNT(20)/1.0,2.0,3.0,4.0,5.0,6.0,7.0,8.0,9.0,10.0,11.0,12.0,
000014       026          *13.0,14.0,15.0,16.0,17.0,18.0,19.0,20.0/
000015       026   C
000016       030   C      READ CLUSTER NUMBER DESIRED AND FLAG FOR REORDERING VARIATES.
000017       030   C      THESE PARAMETERS ARE ENTERED BY THE USER IN BATCH MODE OR AT
000018       030   C       THE TERMINAL IN FREE FORMAT.
000019       030   C
000020       026           READ *,ICLUST,IFLAG
000021       026           IF(ICLUST.LT.2.OR.ICLUST.GT.3) STOP 'WRONG NUMBER OF CLUSTERS'
000022       026   C
000023       026   C     READ VARIABLES ON UNIT 12
000024       026   6       READ(12,100) NVAR,NCLUST
000025       026   100     FORMAT(2I5)
000026       026           IF(NCLUST.GT.3.OR.NVAR.GT.20) GO TO 9999
000027       030   C
000028       030   C       P=PROPORTION IN EACH CLUSTER.
000029       030   C       ABOUND(I,1)=UPPER 3-SIGMA LIMIT FOR THE ITH VARIATE FOR
000030       030   C       ENTIRE DATA SET.
000031       030   C       ABOUND(I,2)=LOWER 3-SIGMA LIMIT FOR THE ITH VARIATE FOR
000032       030   C       ENTIRE DATA SET.
000033       030   C       DM=MEAN VALUE FOR UNTRANSFORMED VARIABLE OR GEOMETRIC MEAN
000034       030   C       FOR TRANSFORMED VARIABLE.
000035       030   C       DU=UPPER 2-SIGMA CLUSTER LIMIT.
000036       030   C       DL=LOWER 2-SIGMA CLUSTER LIMIT.
000037       030   C       (I,J)=JTH CLUSTER, ITH VARIATE.
000038       030   C
000039       026           READ(12,200,END=8888) (P(I),I=1,NCLUST)
000040       029   200     FORMAT(10F5.3)
000041       026           DO 5 J=1,2
000042       026           READ(12,300,END=8888) (ABOUND(I,J),I=1,NVAR)
000043       026   300     FORMAT(10E12.5)
000044       026   5        CONTINUE
000045       026           READ(12,400,END=8888) ((DM(I,J),DU(I,J),DL(I,J),J=1,NCLUST),
000046       026          *I=1,NVAR)
000047       026   400     FORMAT(3E20.5)
```

```
000048    026          IF (NCLUST.EQ.ICLUST) GO TO 7
000049    026          GO TO 6
000050    026     7    IF(NCLUST.GT.2) GO TO 9
000051    026          IF(P(1).GE.P(2)) IORD=1
000052    026          IF(P(2).GT.P(1)) IORD=2
000053    026          GO TO 12
000054    026     9    IF(P(1).GE.P(2).AND.P(1).GE.P(3)) IORD=1
000055    026          IF(P(2).GE.P(1).AND.P(2).GE.P(3)) IORD=2
000056    026          IF(P(3).GE.P(2).AND.P(3).GE.P(1)) IORD=3
000057    030     C
000058    030     C    IF IFLAG=0,VARIATES RETAIN THEIR ORIGINAL ORDER.
000059    030     C    IF IFLAG=A NONZERO INTEGER, VARIATES ARE REORDERED SO THAT
000060    030     C    THE CURVE FOR THE LARGEST CLUSTER RISES ACROSS THE CHART,
000061    030     C    VIZ, INCREASES MOMOTONICALLY.
000062    030     C
000063    026     12   IF(IFLAG.EQ.0) GO TO 8
000064    026          NTOP=NVAR-1
000065    026          DO 10 I=1,NTOP
000066    026          NEXT=I+1
000067    026          DO 11 IP=NEXT,NVAR
000068    026          IF(DM(I,IORD).LE.DM(IP,IORD)) GO TO 11
000069    026          DO 15 IVAR=1,NCLUST
000070    026          A=DM(I,IVAR)
000071    026          DM(I,IVAR)=DM(IP,IVAR)
000072    026          DM(IP,IVAR)=A
000073    026          A=DL(I,IVAR)
000074    026          DL(I,IVAR)=DL(IP,IVAR)
000075    026          DL(IP,IVAR)=A
000076    026          A=DU(I,IVAR)
000077    026          DU(I,IVAR)=DU(IP,IVAR)
000078    026          DU(IP,IVAR)=A
000079    026     15   CONTINUE
000080    026          A=ABOUND(I,1)
000081    026          ABOUND(I,1)=ABOUND(IP,1)
000082    026          ABOUND(IP,1)=A
000083    026          A=ABOUND(I,2)
000084    026          ABOUND(I,2)=ABOUND(IP,2)
000085    026          ABOUND(IP,2)=A
000086    026          VARLAB(21)=VARLAB(I)
000087    026          VARLAB(I)=VARLAB(IP)
000088    026          VARLAB(IP)=VARLAB(21)
000089    026     11   CONTINUE
000090    026     10   CONTINUE
000091    026          WRITE(10,100) NVAR,NCLUST
000092    026          WRITE(10,200)(P(I),I=1,NCLUST)
000093    026          WRITE(10,300)(ABOUND(I,J),I=1,NVAR)
000094    026          WRITE(6,400)((DM(I,J),DU(I,J),DL(I,J),J=1,NCLUST),I=1,NVAR)
000095    026     C
000096    026     C    INITIALIZE THE PLOTTING ROUTINE
000097    026     8      CALL PLOTS(0,0,14)
000098    026     C    MOVE AWAY FROM BOTTOM OF PLOTTER AND INITIALIZE A NEW ORIGIN
```

```
026         CALL PLOT(1.0,1.0,-3)
026   C
026   C   DRAW THE THREE LINES
026         XLEN=NVAR * 1.0
026         CALL PLOT(1.0,2.0,3)
026         CALL PLOT(XLEN,2.0,2)
026         CALL PLOT(XLEN,4.5,3)
026         CALL PLOT(1.0,4.5,2)
026         CALL PLOT(1.0,7.0,3)
026         CALL PLOT(XLEN,7.0,2)
026   C
026   C   PLOT VARIABLE LABELS AND BOUNDS ON VARIABLES
026         XPT=0.0
026         DO 1 I=1,NVAR
026         XPT=XPT+1.0
026         IF(I.LE.9) CALL SYMBOL(XPT-.2,1.25,.14,VARLAB(I),0.0,2)
026         IF(I.GT.9) CALL SYMBOL(XPT-.10,1.25,.14,VARLAB(I),0.0,2)
026         CALL NUMBER(XPT-.3,1.75,.07,ABOUND(I,2),0.0,4)
026         CALL SYMBOL(XPT,2.00,.07,13,0.0,-1)
026         CALL NUMBER(XPT-.3,7.25,.07,ABOUND(I,1),0.0,4)
026   1     CONTINUE
026   C
026   C   PLOT TITLE AND LABELS
026         TITLOC=((NVAR-2.0) * .5)+.50
026         CALL SYMBOL(TITLOC,.5,.21,TITLE,0.0,12)
026         CALL SYMBOL(0.0,7.0,.07,PLUS,0.0,4)
026         CALL SYMBOL(0.0,7.07,.07,BAR,0.0,1)
026         CALL SYMBOL(0.0,4.5,.07,MEAN,0.0,6)
026         CALL SYMBOL(0.0,4.57,.07,BARS,0.0,6)
026         CALL SYMBOL(.40,4.45,.05,G,0.0,1)
026         CALL SYMBOL(0.0,2.0,.07,MINUS,0.0,4)
026         CALL SYMBOL(0.0,2.07,.07,BAR,0.0,1)
026   C
026   C   LOCATE ORIGIN AT OVERALL MEAN AND PLOT PROPORTIONS AND PROFILE VALUES
026         CALL PLOT(0.0,4.5,-3)
026         DO 2 J=1,NCLUST
026         IF(J.EQ.1) CALL NEWPEN(2)
026         IF(J.EQ.2) CALL NEWPEN(3)
026         IF(J.EQ.3) CALL NEWPEN(4)
026         IF(J.EQ.1) ISYM=11
026         IF(J.EQ.2) ISYM=5
026         IF(J.EQ.3) ISYM=1
026         DO 3 I=1,NVAR
026         PMEAN = DM(I,J) * 2.5
026         IF(I.EQ.1) CALL NUMBER(.5,PMEAN,.07,P(J),0.0,3)
026         IF(I.EQ.1) CALL PLOT(COUNT(I),PMEAN,3)
026         IF(I.EQ.1) CALL SYMBOL(COUNT(I),PMEAN,.07,ISYM,0.0,-1)
026          CALL PLOT(COUNT(I),PMEAN,2)
026         IF(I.GT.1) CALL SYMBOL(COUNT(I),PMEAN,.07,ISYM,0.0,-1)
026   3     CONTINUE
```

```
000149       026          DO 4 I=1,NVAR
000150       026          PHIGH=DU(COUNT(I),J)*2.5
000151       026          PLOW=DL(COUNT(I),J)*2.5
000152       026          CALL SYMBOL(COUNT(I),PHIGH,.07,ISYM,0.0,-1)
000153       026          CALL SYMBOL(COUNT(I),PLOW,.07,ISYM,0.0,-1)
000154       026    4     CONTINUE
000155       026    2     CONTINUE
000156       026          CALL PLOT(25.0,0.0,999)
000157       026          STOP
000158       026    8868  STOP 'EOF ENCOUNTERED ON DATA'
000159       026    9999  STOP 'MAXIMUM BOUNDS EXCEEDED - NVAR OR NCLUST'
000160       026          END

END ELT.

@HDG,P  ******  MAPCAL  ******
```

6.4 APPENDIX D. LISTING OF ELLIPSE-PLOTTING ROUTINE

Preceding page blank

```
@ELT,L ELLIPSE.MAIN
ELT 8R1 S74Q1C 12/16/80 15:57:33 (55)
     1.         51              COMMON/CINFO/NVAR,NOBS,OVMEAN(12),OVSTND(12),CORREL(12,12,15),
     2.         51             *CMEAN(12,15),STNDEV(12,15),NCLUST,NTYPE(1000),DATA(1000,12),
     3.         51             *ICVFLG,CPROP(10)
     4.         51              CHARACTER*20 VAR(12)
     5.         51              CHARACTER * 40 TITLE
     6.         51              CHARACTER * 30 CTITLE
     7.         51              DIMENSION PROB(3),XARRAY(1002),YARRAY(1002),ANGLE(15),EIGEN1(15),
     8.         51             *EIGEN2(15)
     9.         51              DATA XLNGTH/14.0/,YLNGTH/10.0/
    10.         51      C
    11.         51      C       THESE PARAMETERS ARE ENTERED BY THE USER AT THE TERMINAL
    12.         51      C       OR IN BATCH WITH A FREE FORMAT.
    13.         51      C
    14.         52      C       ICLUST=NUMBER OF CLUSTERS TO BE PLOTTED - MAXIMIUM NUMBER IS FIVE.
    15.         51      C       IVARS=NUMBER OF VARIATES TO BE PLOTTED.
    16.         51      C       IX=VARIATE NUMBER TO BE PLOTTED ON ABSCISSA.
    17.         51      C       IY=VARIATE NUMBER TO BE PLOTTED ON ORDINATE.
    18.         51      C       IXFLAG=FLAG TO INDICATE IF X-AXISVARIATE IS A WIND COMPONENT.
    19.         51      C       IYFLAG=FLAG TO INDIATE IF Y-AXIS VARIATF IS A WIND COMPONENT.
    20.         51      C             =0 IF NOT A WIND COMPONENT.
    21.         51      C             =1 IF A WIND COMPONENT.
    22.         55      C       IPLOT=FLAG TO INDICATE IF INDIVIDUAL POINTS ARE TO BE PLOTTED
    23.         55      C            EQUALS 1 IF POINTS ARE NOT TO BE PLOTTED
    24.         55      C            DOES NOT EQUAL 1 IF POINTS ARE TO BE PLOTTED - DEFAULT OPTION
    25.         51      C
    26.         55              READ *,ICLUST,IVARS,IX,IY,IXFLAG,IYFLAG,IPLOT
    27.         53              IF(ICLUST.GT.5) STOP ' MAXIMUM BOUNDS EXCEEDED - ICLUST'
    28.         51              DO 200 I=1,IVARS
    29.         51              READ *,VAR(I)
    30.         51      200     CONTINUE
    31.         51      C
    32.         51      C       VAR(I)=CHARACTER STRING OF 20 CHARACTERS OR FEWER DESCRIBING
    33.         51      C       THE_VARIABLE, I.
    34.         51      C       TITLE=CHARACTER STRING OF 40 CHARACTERS OR FEWER FOR PLOT TITLE.
    35.         51      C       CTITLE=CHARACTER STRING OF 30 CHARACTERS OR FEWER FOR
    36.         51      C       CHART TITLE.
    37.         51      C
    38.         51              READ *,TITLE
    39.         51              READ *,CTITLE
    40.         51              CALL READ1
    41.         51              IF(IVARS.NE.NVAR) STOP 'WRONG NUMBER OF VARIABLES'
    42.         51      300     CALL READ2
    43.         51              IF(NCLUST.LT.ICLUST)GO TO 300
    44.         51              IF(NCLUST.GT.ICLUST) STOP 'CLUSTER NUMBER'
    45.         51      C
    46.         51      C       EQUAL CLUSTERS
    47.         51      C
    48.         51              DO 400 I=1,NOBS
    49.         51              XARRAY(I)=DATA(I,IX)
    50.         51              YARRAY(I)=DATA(I,IY)
    51.         51      400     CONTINUE
    52.         51      C
    53.         51      C       NPROB=NUMBER OF ELLIPTIC PROBABILITY CURVES TO BE DRAWN.
    54.         51      C       PROB(I)=ELLIPSE PROBABILITY FOR ITH CURVE.
    55.         51      C         DEFAULT IS ONE ELLIPSE WITH PROBABILITY EQUAL .50.
```

```
51          IF(NPROB.NE.0) GO TO 5
51          NPROB=1
51          PROB(1)=.50
51    C
51          READ *,NPROB,(PROB(I),I=1,NPROB)
51    C
51    C     DETERMINE MAXIMUM PROBABILITY
51    5     GO TO (500,510,520)NPROB
51    500   PRBMAX=PROB(1)
51          GO TO 600
51    510   PRBMAX=AMAX1(PROB(1),PROB(2))
51          GO TO 600
51    520   PRBMAX=AMAX1(PROB(1),PROB(2),PROB(3))
51    600   NPTS=NOBS
51          ILOOP=0
51          IXF=IXFLAG
51          IYF=IYFLAG
51          IF(IXF.EQ.0.AND.IYF.EQ.0) GO TO 700
51          IXF=1
51          IYF=1
51    700   CALL SCFACT(XARRAY,NPTS,XLNGTH,AMINX,DELTAX,IXF)
51          CALL SCFACT(YARRAY,NPTS,YLNGTH,AMINY,DELTAY,IYF)
51          CALL VECTOR(ANGLE,EIGEN1,EIGEN2,DELTAX,DELTAY,IX,IY)
51          CALL XTREME(DELTAX,DELTAY,XMIN,XMAX,YMIN,YMAX,PRBMAX,EIGEN1,
51         1         AMINX,AMINY,IX,IY)
51          IF(XMIN.LT.0.OR.YMIN.LT.0.OR.XMAX.GT.XLNGTH.OR.YMAX.GT.YLNGTH)
51         *GO TO 800
51          GO TO 900
51    C
51    C     COME HERE IF RESCALING IS NEEDED
51    C
51      800 IF(XMIN.LT.0.0) XARRAY(1)=XMIN*DELTAX+AMINX
51          IF(XMIN.GE.0.0) XARRAY(1)=AMINX
51          IF(XMAX.GT.XLNGTH) XARRAY(2)=XMAX*DELTAX+AMINX
51          IF(XMAX.LE.XLNGTH) XARRAY(2)=AMINX+XLNGTH*DELTAX
51          IF(YMIN.LT.0.0) YARRAY(1)=YMIN*DELTAY+AMINY
51          IF(YMIN.GE.0.0) YARRAY(1)=AMINY
51          IF(YMAX.GT.YLNGTH) YARRAY(2)=YMAX*DELTAY+AMINY
51          IF(YMAX.LE.YLNGTH) YARRAY(2)=AMINY+YLNGTH*DELTAY
51    C
51    C     THE FIRST TWO LOCATIONS IN THE ARRAYS HOLD THE EXTREME
51    C     POINTS.  WITH THIS INFORMATION, GO AND RESCALE.
51    C
51          ILOOP=ILOOP+1
51          IPRNT=MOD(ILOOP,5)
51          IF(IPRNT.EQ.0) PRINT *,'ITERATION ',ILOOP
51          IF(ILOOP.EQ.20) PRINT *,'POSSIBLE SCALING PROBLEMS'
51          IF(ILOOP.EQ.20) GO TO 900
51          GO TO 700
51    C
51    C     THE FIRST TWO VALUES IN XARRAY, YARRAY MAY BE ALTERED-RESTORE
51    C
51    900   DO 1000 I=1,2
51          XARRAY(I)=DATA(I,IX)
51          YARRAY(I)=DATA(I,IY)
51    1000  CONTINUE
51    C
```

```
 113.        51     C      INITIALIZE THE PLOTTING ROUTINES
 114.        51     C
 115.        51            CALL PLOTS(0,0,14)
 116.        51     C
 117.        51     C      MOVE AWAY FROM BOTTOM OF PLOTTER AND INITIALIZE A NEW ORIGIN
 118.        51     C
 119.        51            CALL PLOT(1.,1.,-3)
 120.        51            CALL PLOT(0.0,2.0,-3)
 121.        51     C
 122.        51     C      PERFORM PLOTTING FOR A MAXIMUM OF THREE PROBABILITY LEVELS
 123.        51     C
 124.        51            DO 5000 I=1,3
 125.        51            J=(I-1)*5+1
 126.        51            JMAX=I*5
 127.        51            CALL ELPLOT(J,JMAX,XARRAY,YARRAY,IX,IY,VAR,TITLE,PROB,NPROB,
 128.        55           1            EIGEN1,EIGEN2,ANGLE,IXFLAG,IYFLAG,IPLOT)
 129.        51            CALL PLOT(0.,14.,-3)
 130.        51            CALL CHART(J,JMAX,VAR,CTITLE)
 131.        51            CALL PLOT(12.,-14.,-3)
 132.        51            IF(NCLUST.LE.JMAX) GO TO 6000
 133.        51      5000 CONTINUE
 134.        51     C
 135.        51     C      TERMINATE PLOTS
 136.        51     C
 137.        51            CALL PLOT(0.,0.,999)
 138.        51     C
 139.        51     C    THIS PROGRAM WAS WRITTEN BY J. E. STEPHENSON IN DECEMBER,1979.
 140.        51     C      MODIFICATIONS WERE MADE BY C.B.CAMERON.
 141.        51     C
 142.        51      6000 STOP
 143.        51            END

END ELT.  ERRORS: NONE. TIME:  0.628 SEC. IMAGE COUNT: 143
```

```
@ELT,L ELLIPSE.ASCALE
EL1007 SL73R1 02/21/80 15:59:29 (2,)
000001       000          SUBROUTINE ASCALE(VAR,FIRSTV,DELTA)
000002       000    C
000003       000    C     THIS SUBROUTINE TAKES AN UNSCALED VALUE AND SCALES IT
000004       000    C     IN ACCORDANCE WITH THE SCALING PARAMETERS.  THE VALUE
000005       000    C     OF THE VARIABLE IS CHANGED TO THE SCALED VALUE.
000006       000    C
000007       000    C     VAR        - ON INPUT, VALUE TO BE SCALED
000008       000    C                  ON OUTPUT, SCALED VALUE
000009       000    C     FIRSTV     - UNSCALED VALUE AT START OF AXIS
000010       000    C     DELTA      - SCALE FACTOR IN UNITS/INCH
000011       000    C
000012       000          VAR=(VAR-FIRSTV)/DELTA
000013       000          RETURN
000014       000          END

END ELT.
```

```
@ELT,L ELLIPSE.READ1
ELT 8R1 S74Q1C 10/14/80 10:24:27 (18)
     1.          14            SUBROUTINE READ1
     2.          14            COMMON/CINFO/NVAR,NOBS,OVMEAN(12),OVSTND(12),CORREL(12,12,15),
     3.          14           *CMEAN(12,15),STNDEV(12,15),NCLUST,NTYPE(1000),DATA(1000,12),
     4.          14           ICVFLG,CPROP(10)
     5.          14      C
     6.          14      C     NVAR=NUMBER OF VARIABLES IN STORED DATA SET.
     7.          14      C     NOBS=NUMBER OF OBSERVATIONS IN STORED DATA SET.
     8.          14      C     DATA(I,J)=ITH OBSERVATION OF JTH VARIABLE.
     9.          14      C     OVMEAN(I)=MEAN VALUE (OR GEOMETRIC MEAN FOR TRANSFORMED DATA)
    10.          14      C     OF ITH VARIATE FOR THE ENTIRE DATA SET.
    11.          14      C     OVSTND(I)=STANDARD DEVIATION OF ITH VARIATE FOR THE
    12.          14      C     ENTIRE DATA SET.
    13.          14      C
    14.          14            READ(10,100)NVAR,NOBS
    15.          14      100   FORMAT(2I10)
    16.          15            IF(NVAR.GT.12.OR.NOBS.GT.1000) GO TO 9999
    17.          14            DO 300 I=1,NOBS
    18.          14            READ(10,200,END=8888) (DATA(I,J),J=1,NVAR)
    19.          17      200   FORMAT(6F12.5,/,6E12.5)
    20.          14      300   CONTINUE
    21.          14            READ(10,200,END=8888) (OVMEAN(I),I=1,NVAR)
    22.          14            READ(10,200,END=8888) (OVSTND(I),I=1,NVAR)
    23.          14            RETURN
    24.          14      8888  STOP 'EOF ENCOUNTERED ON DATA'
    25.          14      9999  STOP 'MAXIMUM BOUNDS EXCEEDED - NVAR OR NOBS'
    26.          14            END

END ELT.  ERRORS: NONE. TIME: 0.869 SEC. IMAGE COUNT: 26

@FIN
```

```
@ELT,L ELLIPSE.READ2
ELT 8R1 S74Q1C 10/14/80 10:24:02 (24)
    1.           20              SUBROUTINE READ2
    2.           20              COMMON/CINFO/NVAR,NOBS,OVMEAN(12),OVSTND(12),CORREL(12,12,15),
    3.           20             *CMEAN(12,15),STNDEV(12,15),NCLUST,NTYPE(1000),DATA(1000,12),
    4.           20             *ICVFLG,CPROP(10)
    5.           20              DIMENSION DUMMY(12)
    6.           20        C
    7.           20        C     NCLUST=NUMBER OF CLUSTERS STORED IN DATA SET.
    8.           20        C     ICVFLG=CORRELATION MATRIX SWITCH.
    9.           21        C           =0 FOR IDENTICAL CORRELATION MATRICES AMONG CLUSTERS.
   10.           20        C           =1 FOR DIFFERENT CORRELATION MATRICES AMONG CLUSTERS.
   11.           20        C     CPROP(I)=PROPORTION OF DATA IN ITH CLUSTER.
   12.           20        C     CMEAN(J,I)=MEAN VALUE (OR GEOMETRIC MEAN FOR TRANSFORMED
   13.           20        C     DATA) FOR JTH VARIATE AND ITH CLUSTER.
   14.           20        C     STNDEV(J,I)=STANDARD DEVIATION FOR JTH VARIATE AND ITH CLUSTER
   15.           20        C     CORREL(K,J,I)=CORRELATION COEFFICIENT FOR JTH AND KTH
   16.           20        C     VARIATES AND ITH CLUSTER.
   17.           20        C     NTYPE(I)=CLUSTER MEMBERSHIP INDICATOR FOR ITH OBSERVATION.
   18.           20        C     DUMMY(J)=DUMMY READ FOR JTH VARIABLE.
   19.           20        C     IF THE FINAL DATA SET IS BEING PROCESSED, 2 BLANK LINES
   20.           20        C     OR CARDS ARE REQUIRED AT THIS POINT.
   21.           20        C
   22.           20              READ(10,100,END=9999)NCLUST,ICVFLG
   23.           24              IF(NCLUST.GT.5)GO TO 8888
   24.           20        100   FORMAT(2I5)
   25.           20              READ(10,800) (CPROP(I),I=1,NCLUST)
   26.           20        800   FORMAT(10(1X,A4))
   27.           20              DO 200 I=1,NCLUST
   28.           20              READ(10,300,END=9999) (CMEAN(J,I),J=1,NVAR)
   29.           23        300   FORMAT(6F12.5,/,6F12.5)
   30.           20              READ(10,300,END=9999) (STNDEV(J,I),J=1,NVAR)
   31.           20              DO 400 J=1,NVAR
   32.           20              READ(10,300,END=9999) (CORREL(K,J,I),K=1,NVAR)
   33.           20        400   CONTINUE
   34.           20        200   CONTINUE
   35.           20              DO 500 I=1,NOBS
   36.           20              READ(10,600,END=9999)NTYPE(I)
   37.           20        600   FORMAT(10X,I10)
   38.           20        500   CONTINUE
   39.           20              DO 700 I=1,2
   40.           20              READ(10,300,END=9999) (DUMMY(J),J=1,NVAR)
   41.           20          700 CONTINUE
   42.           20              RETURN
   43.           20        9999  STOP'EOF ENCOUNTERED ON DATA'
   44.           20        8888  STOP'MAXIMUM ROUNDS EXCEEDED - NCLUST'
   45.           20              END
END ELT.  ERRORS: NONE. TIME: 0.965 SEC. IMAGE COUNT: 45

@FIN
```

```
@ELT,L ELLIPSE.CHART
ELT 8R1 S74Q1C 11/14/80 14:23:36 (17)
        13          SUBROUTINE CHART(J,JMAX,VAR,CTITLE)
        13     C
        13     C    THIS SUBROUTINE PRODUCES THE CHARTS ASSOCIATED WITH THE ELLIPSE
        13     C    PLOTS
        13     C
        13     C    J           - THE NUMBER OF THE FIRST CLUSTER TO BE INCLUDED IN
        13     C                  CHART
        13     C    JMAX        - THE MAXIMUM NUMBER OF THE LAST CLUSTER TO BE INCLUDED
        13     C                  IN THE CHART.  PLOTTING STOPS WITH THE CLUSTER WHOSE
        13     C                  NUMBER IS THE MINIMUM OF JMAX AND NCLUSTER.
        13     C    VAR         - ARRAY OF ALPHANUMERIC CHARACTER STRINGS TO BE USED
        13     C                  IN PLOTTING THE TABLE.
        13     C
        13           COMMON/CINFO/NVAR,NOBS,OVMEAN(12),OVSTND(12),CORREL(12,12,15),
        13          *CMEAN(12,15),STNDEV(12,15),NCLUST,NTYPE(1000),DATA(1000,12),
        13          *ICVFLG,CPROP(10)
        13           CHARACTER*2 NUM(15)/' 1',' 2',' 3',' 4',' 5',' 6',' 7',' 8',' 9',
        13          *'10','11','12','13','14','15'/
        14           CHARACTER*14 CLUST/'CLUSTER    N = '/,VARBLE/'VARIABLE'/
        13           CHARACTER*4 MEAN/'MEAN'/
        13           CHARACTER*18 STND/'STANDARD DEVIATION'/
        13           CHARACTER * 30 CTITLE
        13           CHARACTER*20 VAR(12)
        13           CHARACTER * 2 PRO/'P='/
        13     C
        13     C     PRODUCE THE FORM
        13     C
        13           CALL PLOT(0.0,0.0,3)
        13           CALL PLOT(0.0,10.0,2)
        13           CALL PLOT(8.0,10.0,2)
        13           CALL PLOT(8.0,0.0,2)
        13           CALL PLOT(0.0,0.0,2)
        13     C
        13     C     INSIDE LINES
        13     C
        13           CALL PLOT(0.0,9.7,3)
        13           CALL PLOT(8.0,9.7,2)
        13           CALL PLOT(8.0,9.6,3)
        13           CALL PLOT(0.0,9.6,2)
        13           CALL PLOT(3.0,9.6,3)
        13           CALL PLOT(3.0,9.3,2)
        13           CALL PLOT(8.0,9.3,2)
        13     C
        13     C     DETERMINE MAXIMUM NUMBER OF COLUMNS
        13     C
        13           IMAX1=JMAX-J+1
        13           IMAX2=NCLUST-J+1
        13           IMAX=MIN(IMAX1,IMAX2)
        13           IDELTA=IMAX-1
        13           JBIG=J+IDELTA
        13     C
        13     C     DETERMINE MAXIMUM X-LIMIT
        13     C
        13           XMAX=3.0+FLOAT(IMAX)*1.00
        13           CALL PLOT(XMAX,9.0,3)
```

```
  56.          13               CALL PLOT(0.0,9.0,2)
  57.          13       C
  58.          13               CALL PLOT(3.0,8.75,3)
  59.          13               CALL PLOT(XMAX,8.75,2)
  60.          13               XLINE=3.1
  61.          13               DO 1 I=1,NCLUST
  62.          13               CALL SYMBOL(XLINE,8.8,.14,PRO,0.0,2)
  63.          13               XLINE =XLINE+.35
  64.          13               CALL SYMBOL (XLINE,8.8,.14,CPROP(I),0.0,4)
  65.          13               XLINE=XLINE+.65
  66.          13       1       CONTINUE
  67.          13               CALL PLOT(0.0,9.0,3)
  68.          13       C
  69.          13       C       NVAR+1 HORIZONTAL LINES ARE TO BE DRAWN
  70.          13       C
  71.          13               NVAR1=NVAR+1
  72.          13               DO 100 I=1,NVAR1
  73.          13               YLOC=9.0-FLOAT(I-1)*0.75
  74.          13               CALL PLOT(0.0,YLOC,3)
  75.          13               CALL PLOT(XMAX,YLOC,2)
  76.          13       100     CONTINUE
  77.          13               YMIN=YLOC
  78.          13       C
  79.          13       C       DRAW VERTICAL LINES
  80.          13       C
  81.          13               DO 200 I=1,IMAX+1
  82.          13               XLOC=3.0+FLOAT(I-1)*1.0
  83.          13               CALL PLOT(XLOC,9.3,3)
  84.          13               CALL PLOT(XLOC,YMIN,2)
  85.          13       200     CONTINUE
  86.          13       C
  87.          13       C       TITLE
  88.          13       C
  89.          13               CALL SYMBOL(1.60,9.75,0.21,CTITLE,0.0,30)
  90.          13       C
  91.          13       C       HEADINGS
  92.          13       C
  93.          13               CALL SYMBOL(0.90,9.05,0.14,VARBLE,0.0,8)
  94.          15               CALL SYMBOL(4.00,9.35,0.14,CLUST,0.0,14)
  95.          17               FNOBS=NOBS*1.0
  96.          17               CALL NUMBER(9.9,9.35,0.14,FNOBS,0.0,-1)
  97.          13               DO 300 I=1,NVAR
  98.          13               YLOC=8.75-FLOAT(I-1)*0.75
  99.          13               CALL SYMBOL(0.10,YLOC,0.14,VAR(I),0.0,20)
 100.          13       300     CONTINUE
 101.          13               DO 350 I=J,JBIG
 102.          13               XLOC=3.35+FLOAT(I-J)*1.0
 103.          13               CALL SYMBOL(XLOC,9.05,0.14,NUM(I),0.0,2)
 104.          13           350 CONTINUE
 105.          13       C
 106.          13       C       OTHER INFO
 107.          13       C
 108.          13               DO 400 I=1,NVAR
 109.          13               YLOC=8.55-FLOAT(I-1)*0.75
 110.          13               CALL SYMBOL(0.38,YLOC,0.14,MEAN,0.0,4)
 111.          13               DO 500 II=J,JBIG
 112.          13               XLOC=3.10+FLOAT(II-J)*1.0
```

```
   113.          13              CALL  NUMBER(XLOC,YLOC,0.14,CMEAN(I,II),0.0,2)
   114.          13       500    CONTINUE
   115.          13              YLOC=YLOC-0.20
   116.          13              CALL  SYMBOL(0.38,YLOC,0.14,STND,0.0,18)
   117.          13              DO 600 II=J,JFIG
   118.          13              XLOC=3.10+FLOAT(II-J)*1.0
   119.          13              CALL  NUMBER(XLOC,YLOC,0.14,STNDEV(I,II),0.0,2)
   120.          13       600    CONTINUE
   121.          13       400    CONTINUE
   122.          13              RETURN
   123.          13              END

 END ELT.  ERRORS: NONE. TIME:  1.094 SEC. IMAGE COUNT: 123

 @FIN
```

```
@ELT,L ELLIPSE.ELIPSE
EL1007 SL73R1 02/21/80 15:59:31 (7,)
000001      003        SUBROUTINE ELIPSE(XCENTR,YCENTR,SMAJOR,SMINOR,ANGLE)
000002      003   C
000003      003   C    THIS SUBROUTINE PLOTS A SINGLE ELLIPSE GIVEN THE SCALED
000004      003   C    COORDINATES OF THE CENTER, THE LENGTHS OF THE SEMI-MAJOR AND
000005      003   C    SEMI-MINOR AXES, AND THE ANGLE (IN DEGREES) OF THE ELLIPSE'S
000006      003   C    ORIENTATION WITH RESPECT TO THE X-AXIS.
000007      003   C
000008      004   C    XCENTR    - X DISTANCE (IN INCHES) OF THE CENTER OF THE ELLIPSE
000009      004   C                FROM THE X START POINT
000010      004   C    YCENTR    - Y DISTANCE (IN INCHES) OF THE CENTER OF THE ELLIPSE
000011      004   C                FROM THE Y START POINT
000012      004   C    SMAJOR    - LENGTH OF THE SEMI-MAJOR AXIS IN INCHES
000013      004   C    SMINOR    - LENGTH OF THE SEMI-MINOR AXIS IN INCHES
000014      004   C    ANGLE     - COUNTER-CLOCKWISE ANGLE OF ROTATION (IN DEGREES)
000015      004   C                OF THE EIGENVECTOR CORRESPONDING TO THE LARGER
000016      004   C                EIGENVALUE TO THE POSITIVE X-AXIS
000017      004   C
000018      003        DATA PI/3.1415926/,RAD/57.295779/
000019      003   C
000020      003   C    CALCULATE THE STARTING COORDINATES FOR THE ELLIPSE
000021      003   C
000022      004        XSTART=XCENTR+COS(ANGLE/RAD)*SMAJOR
000023      004        YSTART=YCENTR+SIN(ANGLE/RAD)*SMAJOR
000024      003   C
000025      003   C    PEN IS UP FOR MOVE TO STARTING COORDINATES
000026      003   C
000027      003        IPEN=3
000028      003        THO=0.0
000029      003        THF=360.0
000030      003   C
000031      003   C    PLOT THE ELLIPSE
000032      003   C
000033      003        CALL ELIPS(XSTART,YSTART,SMAJOR,SMINOR,ANGLE,THO,THF,IPEN)
000034      003        RETURN
000035      003        END

END ELT.
```

```
@ELT,L ELLIPSE.ELPLOT
ELT 8R1 S74Q1C 12/30/80 14:43:07 (6)
     1.          02             SUBROUTINE ELPLOT( J,JMAX,XARRAY,YARRAY,IX,IY, VAR,TITLE,PROB,NPROB,
     2.          05           1                           EIGEN1,EIGEN2,ANGLE,IXFLAG,IYFLAG,IPLOT)
     3.          02             COMMON/CINFO/NVAR,NOBS,OVMEAN(12),OVSTND(12),CORREL(12,12,15),
     4.          02            *CMEAN(12,15),STNDEV(12,15),NCLUST,NTYPE(1000),DATA(1000,12),
     5.          02            *ICVFLG,CPROP(10)
     6.          02             CHARACTER*1 NUM(15)/'1','2','3','4','5','6','7','8','9','0',
     7.          02           1                     '1','2','3','4','5'/
     8.          02             CHARACTER * 40 TITLE
     9.          02             CHARACTER*20 VAR(12)
    10.          02             DIMENSION XARRAY(1002),YARRAY(1002),PROB(3),EIGEN1(15),EIGEN2(15)
    11.          02             DIMENSION ANGLE(15),AMEANX(20),AMEANY(20)
    12.          02       C
    13.          02       C     STORE THE CLUSTER MEANS FOR THE X AND Y AXES
    14.          02       C
    15.          02             DO 50 I=1,NCLUST
    16.          02             AMEANX(I)=CMEAN(IX,I)
    17.          02             AMEANY(I)=CMEAN(IY,I)
    18.          02          50 CONTINUE
    19.          02       C
    20.          02       C     DRAW THE X AND Y AXES
    21.          02       C
    22.          02             NOB1=NOBS+1
    23.          02             NOB2=NOBS+2
    24.          02             CALL AXIS(0.0,0.0,VAR(IX),-20,14.0,0.0,XARRAY(NOB1),XARRAY(NOB2))
    25.          02             CALL AXIS(0.0,0.0,VAR(IY),20,10.0,90.0,YARRAY(NOB1),YARRAY(NOB2))
    26.          02       C
    27.          02       C DRAW CENTRAL LINES AT MEAN OR ZERO FOR EASE OF VIEWING
    28.          02       C
    29.          02             IF(IXFLAG.EQ.1) XPT=0.0
    30.          02             IF(IXFLAG.EQ.0) XPT=OVMEAN(IX)
    31.          02             IF(IYFLAG.EQ.1) YPT=0.0
    32.          02             IF(IYFLAG.EQ.0) YPT=OVMEAN(IY)
    33.          02             CALL ASCALE(XPT,XARRAY(NOB1), XARRAY(NOB2))
    34.          02             CALL ASCALE(YPT,YARRAY(NOB1), YARRAY(NOB2))
    35.          02             CALL PLOT(XPT-1.0,YPT,3)
    36.          02             CALL PLOT(XPT+1.0,YPT,2)
    37.          02             CALL PLOT(XPT,YPT-1.0,3)
    38.          02             CALL PLOT(XPT,YPT+1.0,2)
    39.          02       C
    40.          02       C     DETERMINE THE NUMBER OF ELLIPSES TO BE DRAWN
    41.          02       C
    42.          02             IMAX1=JMAX-J
    43.          02             IMAX2=NCLUST-J
    44.          02             IMAX=MIN(IMAX1,IMAX2)
    45.          02             JBIG=J+IMAX
    46.          02       C
    47.          05       C     IF IPLOT IS NOT EQUAL TO 1 - PLOT THE RAW DATA POINTS ON THE GRAPH
    48.          02       C
    49.          06             IF(IPLOT.EQ.1) GO TO 2
    50.          02             DO 100 I=1,NOBS
    51.          02             IJ=NTYPE(I)
    52.          02       C
    53.          02       C     SCALE THE DISTANCES BASED ON PREVIOUSLY CALCULATED
    54.          02       C     SCALING FACTORS
    55.          02       C
```

```
      56.        02             CALL  ASCALE(XARRAY(I),XARRAY(NOB1),XARRAY(NOB2))
      57.        02             CALL  ASCALE(YARRAY(I),YARRAY(NOB1),YARRAY(NOB2))
      58.        02             CALL  SYMBOL(XARRAY(I),YARRAY(I),0.14,NUM(IJ),0.0,1)
      59.        02         100 CONTINUE
      60.        02       C
      61.        02       C     PLOT THE ELLIPSES
      62.        02       C
      63.        05       2     DO 200 I=J,JBIG
      64.        02             RAD1=SQRT(EIGEN1(I))
      65.        02             RAD2=SQRT(EIGEN2(I))
      66.        02             CALL  ASCALE(AMEANX(I),XARRAY(NOB1),XARRAY(NOB2))
      67.        02             CALL  ASCALE(AMEANY(I),YARRAY(NOB1),YARRAY(NOB2))
      68.        02             DO 200 K=1,NPROB
      69.        03             CHISQ=2.0 * LOG(1.0/(1.0-PROB(K)))
      70.        02             CHI=SQRT(CHISQ)
      71.        02             SMAJOR=CHI*RAD1
      72.        02             SMINOR=CHI*RAD2
      73.        02             CALL  ELIPSE(AMEANX(I),AMEANY(I),SMAJOR,SMINOR,ANGLE(I))
      74.        02         200 CONTINUE
      75.        02             CALL  PLOT(0.0,-2.0,-3)
      76.        02             CALL  SYMBOL(0.0,1.0,.14,TITLE,0.0,40)
      77.        02             RETURN
      78.        02             END

 END ELT.  ERRORS: NONE. TIME:  0.986 SEC. IMAGE COUNT: 78

 @FIN
```

```
@ELT,L ELLIPSE.SCFACT
EL1007 SL73R1 02/21/80 15:59:39 (7,)
000001       006          SUBROUTINE SCFACT(ARRAY,NARRAY,ALNGTH,AMIN,DELTA,IFLAG)
000002       003    C
000003       003    C     THIS SUBROUTINE CALCULATES THE SCALING FACTOR (UNITS/INCH)
000004       003    C     WHEN PASSED THE ARRAY NAME, THE NUMBER OF VARIABLES TO BE
000005       003    C     SCANNED, AND THE LENGTH OF THE SCALE IN INCHES.
000006       003    C
000007       003    C     ARRAY     - START ADDRESS OF ARRAY TO BE SCALED
000008       003    C     NARRAY    - NUMBER OF ARRAY VALUES TO BE SEARCHED
000009       003    C     ALNGTH    - LENGTH IN INCHES TO WHICH THE DATA IS TO
000010       003    C                 BE SCALED
000011       003    C     AMIN      - UNSCALED VALUE FOR FIRST POINT ON AXIS
000012       003    C     DELTA     - SCALING FACTOR
000013       003    C
000014       007          DIMENSION ARRAY(352)
000015       003          INC=1
000016       006          IF(IFLAG.EQ.1) INC=-1
000017       003          CALL SCALE(ARRAY,ALNGTH,NARRAY,INC)
000018       003    C
000019       003    C     RETURN START VALUE AND SCALING FACTOR
000020       003    C
000021       003          AMIN=ARRAY(NARRAY+1)
000022       003          DELTA=ARRAY(NARRAY+2)
000023       003          RETURN
000024       003          END

END ELT.
```

```
@ELT,L ELLIPSE.VECTOR
EL1007 SL73R1 02/21/80 15:59:40 (12,)
000001     008        SUBROUTINE VECTOR(ANGLE,EIGEN1,EIGEN2,DELTAX,DELTAY,IX,IY)
000002     008    C
000003     008    C   GIVEN THE CORRELATION COEFFICIENT,STANDARD DEVIATIONS, AND
000004     008    C   SCALING FACTORS, THIS SUBROUTINE CALCULATES THE ANGLE OF
000005     008    C   ROTATION (COUNTER-CLOCKWISE) FROM THE X-AXIS TO THE EIGEN-
000006     008    C   VECTOR CORRESPONDING TO THE LARGER EIGENVALUE.  THE SCALED
000007     008    C   EIGENVALUES ARE RETURNED IN THE ARRAYS EIGEN1 (LARGER EIGEN-
000008     008    C   VALUE) AND EIGEN2 (SMALLER EIGENVALUE).
000009     008    C
000010     008    C   ANGLE     - THE CALCULATED ANGLE OF ROTATION FROM THE X-AXIS
000011     008    C               TO THE EIGENVECTOR WITH THE LARGER EIGENVALUE.
000012     008    C               ANGLE IS IN DEGREES.
000013     008    C   EIGEN1    - THE SCALED (IN INCHES) VALUE OF THE LARGER EIGEN-
000014     008    C               VALUE.
000015     008    C   EIGEN2    - THE SCALED (IN INCHES) VALUE OF THE SMALLER
000016     008    C               EIGENVALUE.
000017     008    C   DELTAX    - THE SCALING FACTOR FOR THE X-AXIS VARIABLE IN
000018     008    C               UNITS/INCH.
000019     008    C   DELTAY    - THE SCALING FACTOR FOR THE Y-AXIS VARIABLE IN
000020     008    C               UNITS/INCH.
000021     008    C   IX        - THE NUMBER WITHIN THE VARIABLE LIST OF THE VARIABLE
000022     008    C               TO BE PLOTTED ON THE X-AXIS.
000023     008    C   IY        - THE NUMBER WITHIN THE VARIABLE LIST OF THE VARIABLE
000024     008    C               TO BE PLOTTED ON THE Y-AXIS.
000025     008    C
000026     008        COMMON/CINFO/NVAR,NOBS,OVMEAN(12),OVSTND(12),CORREL(12,12,15),
000027     012       1          CMEAN(12,15),STNDEV(12,15),NCLUST,NTYPE(350),
000028     012       2          DATA(350,12),ICVFLG,CPROP(10)
000029     008        DIMENSION ANGLE(15),EIGEN1(15),EIGEN2(15)
000030     008        DO 1000 I=1,NCLUST
000031     008    C
000032     008    C    DETERMINE APPROPRIATE STANDARD DEVIATIONS AND SCALE
000033     008    C
000034     008        SIGMAX=STNDEV(IX,I)/DELTAX
000035     008        SIGMAY=STNDEV(IY,I)/DELTAY
000036     008    C
000037     008    C    DETERMINE APPROPRIATE CORRELATION COEFFICIENT
000038     008    C
000039     008        RXY=CORREL(IX,IY,I)
000040     008    C
000041     008    C    BEYOND THE FIRST PASS IF EQUAL COVARIANCE OPTION IS IN
000042     008    C    EFFECT, FURTHER CALCULATIONS CAN BE SKIPPED.
000043     008    C
000044     008        IF(I.EQ.1.OR.ICVFLG.EQ.1) GO TO 1125
000045     008    C
000046     008    C    EQUAL COVARIANCES
000047     008    C
000048     008        ANGLE(I)=ANGLE(I-1)
```

```
000049      008          EIGEN1(I)=EIGEN1(I-1)
000050      008          EIGEN2(I)=EIGEN2(I-1)
000051      008          GO TO 1000
000052      008     C
000053      008     C       UNEQUAL COVARIANCES
000054      008     C
000055      008      1125 ANUM=2.0*RXY*SIGMAX*SIGMAY
000056      008          DENOM=SIGMAX*SIGMAX-SIGMAY*SIGMAY
000057      008          DENABS=ABS(DENOM)
000058      008          IF(DENABS.LT.1.0E-06) DENOM=1.0E-06
000059      008          ARG=ANUM/DENOM
000060      008          IF(ABS(ARG).LT.1000.0) GO TO 1250
000061      008          IF(ARG.GT.0.0) ANGLE(I)=45.0
000062      008          IF(ARG.LE.0.0) ANGLE(I)=-45.0
000063      008          GO TO 1500
000064      008      1250 ANGLE(I)=57.29578*ATAN(ARG)/2.0
000065      009          IF(ANGLE(I).LT.0.0.AND.RXY.GT.0.0) ANGLE(I)=ANGLE(I)+90.0
000066      009          IF(ANGLE(I).GT.0.0.AND.RXY.LT.0.0) ANGLE(I)=ANGLE(I)-90.0
000067      008     C
000068      008     C       NOW CALCULATE THE EIGENVALUES
000069      008     C
000070      008      1500 SUM=SIGMAX*SIGMAX+SIGMAY*SIGMAY
000071      008          DISCRM=SUM*SUM-4.0*SIGMAX*SIGMAX*SIGMAY*SIGMAY*(1.0-RXY*RXY)
000072      008          IF(DISCRM.GT.0.0) GO TO 1750
000073      008     C
000074      008     C       DISCRIMINANT VALUE IS NEGATIVE
000075      008     C
000076      008          WRITE(6,1625) DISCRM,I
000077      008      1625 FORMAT(1X,'** WARNING ** DISCRIMINANT VALUE OF ',E10.5,
000078      008         1      ' FOR CLUSTER ',I2,' SET EQUAL TO 0')
000079      008          DISCRM=0.0
000080      008      1750 EIGEN1(I)=(SUM+SQRT(DISCRM))/2.0
000081      008          EIGEN2(I)=(SUM-SQRT(DISCRM))/2.0
000082      008      1000 CONTINUE
000083      008          RETURN
000084      008          END

END ELT.
```

```
@ELT,L ELLIPSE.XTREME
ELT007 SL73R1 02/21/80 15:59:41 (10,)
000001       006         SUBROUTINE XTREME(DELTAX,DELTAY,XMIN,XMAX,YMIN,YMAX,
000002       006        1            PRBMAX,EIGEN1,AMINX,AMINY,IX,IY)
000003       006    C
000004       006    C    THIS SUBROUTINE SCALES THE MEAN VALUES OF THE ELLIPSES
000005       006    C    TO THE CURRENT SCALE.  EACH CENTROID IS, IN EFFECT,
000006       006    C    ENCLOSED BY A CIRCLE OF RADIUS CHIMAX*EIGEN1(I).  THIS
000007       006    C    RADIUS IS EQUAL TO THE LENGTH OF THE SEMI-MAJOR AXIS
000008       006    C    OF THE ELLIPSE.  THE EXTREMA OF THESE CIRCLES IN THE X
000009       006    C    AND Y DIRECTIONS ARE CALCULATED AND PASSED BACK TO THE
000010       006    C    CALLING ROUTINE.  THESE VALUES ARE USED BY THE CALLING
000011       006    C    ROUTINE TO DETERMINE IF THE PLOT HAS TO BE RESCALED.
000012       006    C
000013       006         COMMON/CINFO/NVAR,NOBS,OVMEAN(12),OVSTND(12),
000014       006        1        CORREL(12,12,15),CMEAN(12,15),STNDEV(12,15),
000015       010        2        NCLUST,NTYPE(350),DATA(350,12),ICVFLG,CPROP(10)
000016       006         DIMENSION EIGEN1(15)
000017       006    C
000018       006    C    CALCULATE CHI FOR THE MAXIMUM PROBABILITY LEVEL
000019       006    C
000020       006         CHISQ=LOG(1.0/(1.0-PRBMAX))
000021       006         CHI=SQRT(CHISQ)
000022       006    C
000023       006    C    SCALE CENTROID OF FIRST CLUSTER FOR INITIAL VALUES
000024       006    C
000025       006         XMIN=(CMEAN(IX,1)-AMINX)/DELTAX
000026       006         XMAX=XMIN
000027       006         YMIN=(CMEAN(IY,1)-AMINY)/DELTAY
000028       006         YMAX=YMIN
000029       006    C
000030       006    C    SEARCH FOR EXTREMA FOR ALL CLUSTERS
000031       006    C
000032       006         DO 100 I=1,NCLUST
000033       006         RAD1=SQRT(EIGEN1(I))
000034       006         CENTRX=(CMEAN(IX,I)-AMINX)/DELTAX
000035       006         CENTRY=(CMEAN(IY,I)-AMINY)/DELTAY
000036       006         UP=CENTRY+CHI*RAD1
000037       006         DOWN=CENTRY-CHI*RAD1
000038       006         ALEFT=CENTRX-CHI*RAD1
000039       006         RIGHT=CENTRX+CHI*RAD1
000040       006         IF(UP.GT.YMAX) YMAX=UP
000041       006         IF(DOWN.LT.YMIN) YMIN=DOWN
000042       006         IF(ALEFT.LT.XMIN) XMIN=ALEFT
000043       006         IF(RIGHT.GT.XMAX) XMAX=RIGHT
000044       006     100 CONTINUE
000045       006         RETURN
000046       006         END

END ELT.

@ASG,A PROFILE.
```

6.5 APPENDIX E. PARAMETERS AND USER AIDS FOR NORMIX AND ASSOCIATED ROUTINES

The prospective user can refer to this appendix to set up runstreams for the NORMIX, NORMIX-PREP, PROFILE, and ELLIPSE programs. In general, the procedure is to list parameters with their code names, default values, and comments, and then add a few remarks.

6.5.1 NORMIX

Refer to Table E-1 for a listing of NORMIX parameters.

NORMIX output is somewhat sensitive to choice of values for parameters NSAMPL, NTYP, ENMIN, and MAXAV. If NSAMPL is small, e.g., < 50, clustering may be inhibited by the presence of outliers and very small groupings, as pointed out in Section 4.2.5. Experience to date with environmental data reveals a propensity to form one or two clusters when there are fewer than 50 observations, but to readily form four or five clusters when there are more than 100 observations. If an NTYP sequence such as 1, 2, 3, 4, 5... fails to progress by use of the chi-square test, physical or chemical processes may justify a change in the sequence, i.e., 1, 3, 4, 5... whereby not two, but three, clusters are significant compared to a single cluster. If NSAMPL is very small, e.g., < 30, or if the number of variates, NVBLES, is large, ENMIN should not be allowed to default to NVBLES ±1. Instead, ENMIN should be set equal to 2.0 or 3.0 to allow small clusters. If NSAMPL is larger than MAXAV, a change in the order of the input data may produce a small variation in the output mapping/clustering.

TABLE E-1. NORMIX PARAMETERS

Code Name	Name	Default	Comments
NVBLES	number of variates	---	mandatory
NSAMPL	number of observations	---	mandatory
INP	switch for read unit	5	if set = 0, switches to 5 for UNIVAC
NTYP	number of hypothesized clusters to be tested in sequence	---	see example
NOR	switch for covariance matrices	0	0 for equal, 1 for unequal
ENMIN	minimum number of points in a cluster	NVBLES + 1	if set = 0, switches to NVBLES + 1
HTEST	probability level for chi-square test	---	suggest .05
MAXAV	size of subsample used to make initial estimates	150 or NSAMPL	40, 80, 120 are convenient for printing
IPRINT	switch for printing output at each iteration	0	
LTIME	time limit on computations in minutes	600	if set less than run time, the difference is available for printing
ITERX	maximum number of iterations	300	suggest 100
IHIER	switch for printing hierarchical grouping	0	1 for printing 3rd and final iteration result
MERGE	switch for printing merging details	0	1 for merge version of final iteration of hierarchical grouping
IGEN	switch for printing eigenvalues and eigenvectors	0	1 for all iterations

(Continued)

TABLE E-1 (continued)

Code Name	Name	Default	Comments
MTIME	switch for printing time expended in certain operations	0	1 for grouping time and discriminant analysis time
NTIME	switch for cutting off computations and punching a deck of input statistics	0	1 for cut off in accordance with LTIME
IHIS	switch for chi-square normality test and histogram for each cluster variable	0	1 for test and displays
IOVER	switch for overriding certain abortive conditions	0	1 for overriding
IMAP	switch for storing information for ellipse-plotting	0	1 for storing, using unit number 11
IUNITS	switch for printing names and units in the output	0	1 for printing column headings; right-justify
IBACK	switch for computing backtransformation	0	1 for computations
IPOL	identification of nonwind variates to be backtransformed	---	Example: 1, 2, 3 Example: 1, 4, 5, 8
NWIND	switch for backtransforming wind components to speed and direction	0	1 for computations
IWIND	identifier of wind variates to be backtransformed	0	Example: 5,6
IPROF	switch for storing information for profile-plotting	0	1 for storing, using unit number 12
POWER	exponent used for transformation in NORMIX-PREP	0	reciprocal of POWER used in backtransformation

When running the UNIVAC 1110 (1979) NORMIX program, the runstream must have the following sequence:

```
@XQT NORMIX.CLUSTER/CLUSTER
     (2 TITLE LINES)
 $LIST               (note:  indent 1 space)
     (NORMIX parameters)
$END
     (1 identification line for each variable)
     (format for reading data file)
     (blank line or preliminary estimates)
@ADD, P DATA FILE.
@FIN
```

6.5.2 NORMIX-PREP

Refer to Table E-2 for a listing of NORMIX-PREP parameters.

NORMIX-PREP need not be run unless one or more of the following displays or alterations of the original data set is desired:

a. chi-square test for normality
b. equiprobability histogram
c. cumulative distribution plot on normal probability scale
d. natural logarithm transform

TABLE E-2. NORMIX-PREP PARAMETERS

Code Name	Name	Default	Comments
NVBLES	number of variates	---	mandatory
NSAMPL	number of observations	---	mandatory
NPOL	switch for transforming one or more variates	0	1 for transformation
IPOL	list of variate numbers to be converted to natural logarithms or powers	0	Example: 1, 3 Example: 1, 2, 3
NWIND	switch for computing wind components	0	1 for computing
IWIND	list of the 2 variable numbers pertaining to wind speed and wind direction, in that order	0	west wind component and south wind component will replace the wind speed and direction, respectively
FORMTI	format statement for input data	---	
FORMTO	format statement for output data	---	
IHIST	switch for displaying histograms of input data and transformed data	0	calls IMSL Subroutine USHIST when = 1
IRAND	switch for adding a scaled pseudo-random number to each input datum	0	calls IMSL Subroutine GGNQF when = 1
IQOL	list of variate numbers to be altered by a pseudo-random normal deviate when IRAND = 2	0	Example: 1, 3, 4
DEG	number of degrees of a clockwise rotation of wind axes	0	

(continued)

TABLE E-2 (continued)

Code Name	Name	Default	Comments
NCELLS	number of cells for chi-square test for normality and for histograms	---	suggest NSAMPL/5 but no greater than 12
IDF	number of hypothesized distribution function parameters estimated from data	---	generally = 2 (mean and standard deviation)
ISPACE	spacing between histogram bars	---	suggest 4
MINPOL	minimum value desired in input data after scaling	0	floating point; suggest 2.0, e.g., 3*2.0 for 3 variables
THRESH	threshold value, or minimum acceptable input data value	0	if logarithms are to be taken, set $\geq$ 0, e.g., 3*0.0
NCLASS	number of cells for probability charts	---	see NCELLS
IUNIT	unit number for storing output data	10	mandatory
IPROB	switch for displaying probability charts	0	calls CUMPLT Subroutine when = 1
FACTOR	multiplication factor for random number	10^{-5}	if set = 0, switches to 10^{-5}
POWER	power transformation	0	Example: 1/2 for square root

e. power transform

f. pseudo-random normal deviate scaled and added

g. conversion of wind speed and direction to components

h. rotation of wind component axes.

When running the UNIVAC 1110 (1979) NORMIX-PREP program, the runstream must have the following sequence:

```
@XQT NORMIX-PREP.GO
     (2 title lines)
 $LIST              (note:  indent 1 space)
     (NORMIX parameters)
$END
@ADD, P DATAFILE.
@FIN
```

6.5.3 PROFILE

If a profile of variates such as Figure E-1 is to be produced by CALCOMP or other graphics system, the information for plotting is stored by the NORMIX Subroutine called PROFILE. The information is then read by the MAIN program of the external program, PROFILE, in the following sequence, line 1 being supplied by the user and the remaining lines being read from the stored file with an @ADD command:

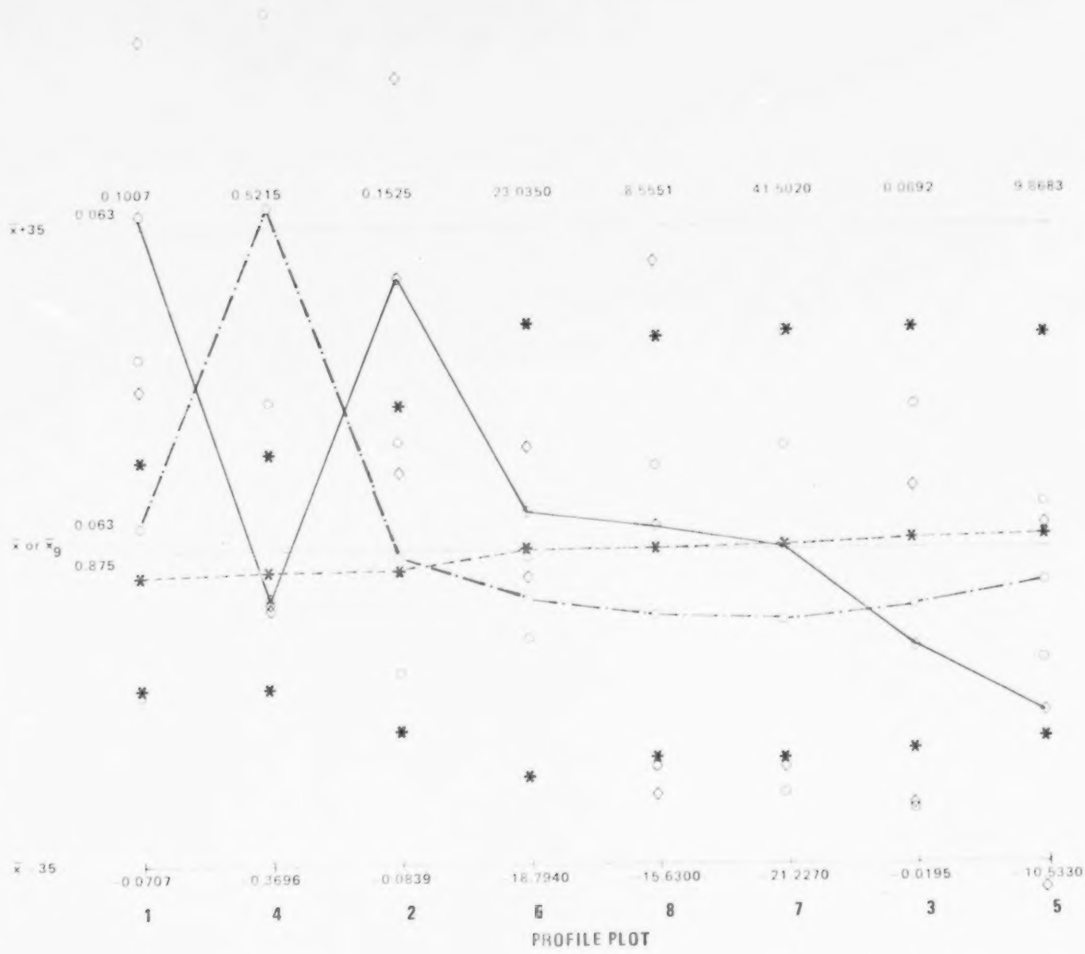


Figure E-1. Cluster variate profile plot for three clusters with eight reordered variates.

a. Line 1 - ICLUST, IFLAG

ICLUST = number of clusters desired, i.e., 2 or 3.

IFLAG = flag for reordering variates.

= 0 if variates are to retain their original order.

≠ 0 if variates are to be reordered so that the curve for the largest cluster rises across the chart, i.e., increases monotonically.

b. Line 2 - NVAR, NCLUST

NVAR = number of variates in data set being read.

NCLUST = number of clusters in data set being read. If this quantity does not equal ICLUST, the data set will not be profiled.

c. Line 3 - P_i, $i = 1, 2, \ldots, n$

P_i = proportion of data in the ith cluster.

d. Line 4 - $ABOUND_{ij}$, $i = 1, 2, \ldots, n; j = 1$

$ABOUND_{ij}$ = upper 3-sigma limit for the ith variate for the entire data set.

e. Line 5 - $ABOUND_{ij}$, $i = 1, 2, \ldots, n; j = 2$

$ABOUND_{ij}$ = lower 3-sigma limit for the ith variate for the entire data set.

f. Line 6 - DM_{ij}, DU_{ij}, DL_{ij}, $i = 1, 2, \ldots, n; j = 1, \ldots$ ICLUST

DM_{ij} = mean value for untransformed variable or geometric mean for transformed variable, for the ith variate and jth cluster.

DU_{ij} = upper 2-sigma limit for the ith variate and jth cluster.

DL_{ij} = lower 2-sigma limit for the ith variate and jth cluster.

The above line numbers are based upon the assumption that no more than 10 variates are included in the data set. If 11 to 20 variates are present, "line 4" becomes "lines 4 and 5," "line 5" becomes "lines 6 and 7," and "line 6" becomes "lines 8 and 9."

6.5.4 ELLIPSE

If ellipses such as those in Figure E-2 are to be drawn by CALCOMP or other graphics system, the data for plotting are stored by the NORMIX program and are read by ELLIPSE with unit number 10. The parameters are entered in the following sequence:

a. Line 1 - ICLUST, IVARS, IX, IY, IXFLAG, IYFLAG, IPLOT

ICLUST = number of clusters to be plotted. The input file is searched until CLUST is located.

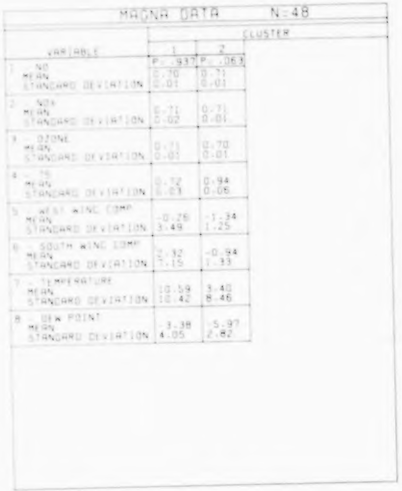
IVARS = number of variates.

IX = variate number to be plotted on x-axis.

IY = variate number to be plotted on y-axis.

IXFLAG = flag for wind component, for the x-axis variate.
= 0 if not a wind component.
= 1 if a wind component.

IYFLAG = flag for wind component for the y-axis variate.
= 0 if not a wind component.
= 1 if a wind component.



MAGNA DATA N=48		
VARIABLE	CLUSTER 1	CLUSTER 2
	P=.937	P=.063
1 - NO MEAN STANDARD DEVIATION	 0.70 0.01	 0.71 0.01
2 - NOX MEAN STANDARD DEVIATION	 0.71 0.02	 0.71 0.01
3 - OZONE MEAN STANDARD DEVIATION	 0.71 0.01	 0.70 0.01
4 - TS MEAN STANDARD DEVIATION	 0.72 0.03	 0.94 0.06
5 - WEST WIND COMP MEAN STANDARD DEVIATION	 -0.26 3.49	 -1.34 1.25
6 - SOUTH WIND COMP MEAN STANDARD DEVIATION	 2.32 7.15	 -0.94 1.33
7 - TEMPERATURE MEAN STANDARD DEVIATION	 10.59 10.42	 3.40 8.46
8 - DEW POINT MEAN STANDARD DEVIATION	 -3.38 4.05	 -5.97 2.82

Figure E-2a.

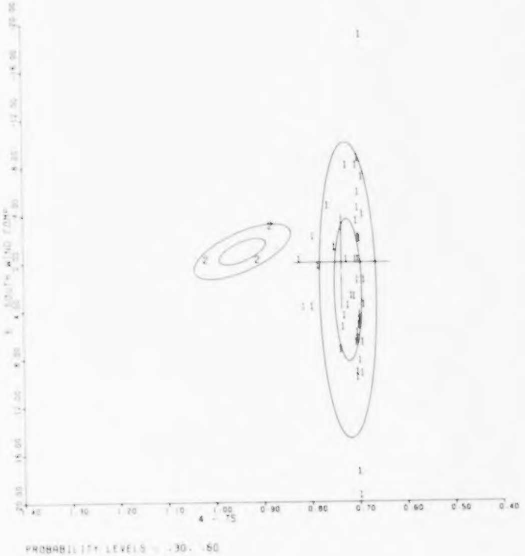


Figure E-2b.

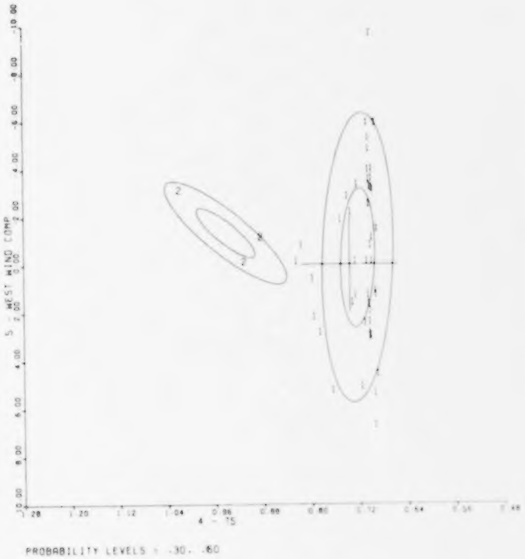


Figure E-2c.

Figure E-2. CALCOMP ellipse plot for two clusters and eight variates.

IPLOT = flag to indicate if individual points are to be plotted.

= 1 if points are not to be plotted.

≠ 1 if points are to be plotted.

The wind component flags are needed to convert their ellipses to conventional meteorological representation.

b. Line 2 - variate description: character string up to 20 characters for variate number 1.

Line 3 - same for variate number 2.

.
.
.

Line IVARS+1 - same for variate number NVARS.

c. Line IVARS+2 - Title of up to 40 characters in a character string, to appear under the plot.

d. Line IVARS+3 - Title of up to 30 characters in a character string, to appear in the data table.

e. LINE IVARS+4 - NPROB, PROB(1), PROB(2), PROB(3)

NPROB = number of probability ellipses to be drawn, but no more than 3.

PROB = probability levels of ellipses to be drawn.

A typical runstream for ELLIPSE is as follows:

```
3, 8, 5, 6, 1, 1
'NO'
'NOX'
'OZONE'
'TS'
'WEST COMP'
'SOUTH COMP'
'T'
'DP'
'WEST VS SOUTH WIND - PROB LEVEL = .50'
'DATA FROM MAGNA - N = 96'
1, .50
```

In its present form, ellipse handles a maximum of 12 variates, 350 observations, and 5 clusters. Two additional blank lines or cards are required at the end of the final set of probability levels, i.e., at the end of the stored data set. These blank lines are read by the "DUMMY READ" statement in ELLIPSE.READ2.

Since wind component scales are reversed in order to be conventional to the viewer, a pollutant mapped against a wind component must also be reversed in scale. The example in figure E-2 shows a conventional plot where west-east wind is on the x-axis with ozone on the ordinate scale reversed, then south-north wind is on the y-axis with ozone on the abscissa, scale reversed.

6.6 APPENDIX F. ILLUSTRATIVE OUTPUTS

To illustrate and explain the output format that is used, a data set is followed throught the four routines with certain options turned on. The data set is from Magna, Utah, a site that is described by Crutcher (1980); the data consist of hourly averages for eight variates at 2000 hours, i.e., 1900 to 2000 local time. The eight variates are No, NO_x, O_3, TS, wind speed, wind direction, temperature, and dew point temperature. The order of presentation is NORMIX-PREP, NORMIX, PROFILE, and ELLIPSE.

6.6.1 NORMIX-PREP

The data on output pages 6-123 to 6-126 were selected from NORMIX-PREP. Page 6-123 lists all of the NAMELIST parametes with their settings. Note that NPOL = 1, indicating that data values falling below THRESH will be increased, for those variates designated by IPOL (i.e., variates 1 to 4). If one value undergoes this upward scaling, all values for the variate are increased a similar amount so that the minimum value for the variate equals MINPOL. Note also that NWIND = 1, indicating that a pair of wind variates need to be converted. Since IWIND = 5, the first wind variate, i.e., speed, is variate number 5. The next variate in line is number 6, and by convention it is the wind direction. After conversion, the west to east wind component resides in number 5 position and the south to north wind component is in number 6 position. Proceeding further, note that IHIST = 1 to demand histogram displays for each variate. However, IRAND = 0, forbidding the addition of scaled pseudo-random numbers. If IRAND were set equal to 1, the variates to be randomized would be designated by IQOL and the multiplicative factor would be set by FACTOR. Finally, IPROB = 1 demands cumulative distribution charts on a probability scale for each variate.

Page 6-124 lists the input data, using an E-format when a value is less than 1.0 X 10^{-1}. Page 6-125 gives an example of an "analysis" of variate number 4 (TS), giving the mean value, standard deviation, cell frequency count, chi-square, and a probability statement on normality. A disparate set of frequencies for the cells and a probability of less than 0.05 support the rejection of the null hypothesis that the distribution is normal. Page 6-125 also displays the optional histogram and cumulative distribution chart for the same variate. These diagrams confirm that TS is not normally distributed. The bulk of the values are or order 10^{-2} or less with 15 percent negative. Because of the negative values, logarithmic transformations cannot be made on the first four variates until the TS values have been scaled upward. If they are scaled upward so that the minimum value is 2, then there is little change in the distribution after transformation. On the other hand, a minimum value in the range of 0 to 1 encourages the formation of singleton clusters on the low end of the data set when logarithms are taken, presenting a dilemma that has not been wholly resolved. Consequently, experimentation by the user would be in order if faced with negative pollutant values. Experience has indicated that a minimum value of 2 gives more normal variate distributions in the clusters.

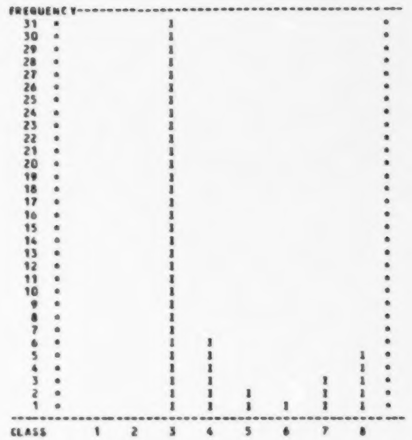
Finally, page 6-126 lists the transformed data. Variates 1 to 4 are the natural logarithms of scaled input values and variates 5 and 6 are the west and south components of the wind, respectively.

```
                                   PROGRAM NORMIX-PREP
                 OPTIONAL NATURAL LOG  TRANSFORMATION, RANDOM NUMBER ADDITION,
               CHI-SQUARE TEST FOR NORMALITY, EQUAL-PROBABILITY INTERVAL HISTOGRAMS,
             PROBABILITY CHARTS, SCALING, DECOMPOSITION OF WIND SPEED INTO COMPONENTS,
                     ROTATION OF WIND AXES, AND VALUES FOR BACKTRANSFORMATION
                   OF TRANSFORMED DATA, ALL OPTIONS AVAILABLE FOR ANY VARIABLE

 NORMIX-PREP PROGRAM WITH MACNA, HOUR 20 DATA

                                                        INPUT PARAMETERS
 $LIST
NVBLES =          8.,NSAMPL =         48.,NPOL =          1.
IPOL =         1.         2.         3.         4.         0.         0.         0.         0.         0.         0.
          0.         0.         0.         0.         0.         0.         0.         0.         0.         0.
NWIND =          0.
IWIND =          0.        0.,FORMTI ='(14X,8F6.0)                                    ',
FORMTO ='(6E12.5,/,6E12.5)                          ',IHIST =          0.,IRAND =          0.
IQOL =         0.         0.         0.         0.         0.         0.         0.         0.         0.         0.
          0.         0.         0.         0.         0.         0.         0.         0.         0.         0.
DEC = .00000000    ,NCELLS =          8.,IDF =          2.,ISPACE =          4.
MINPOL = .20000000+001, .20000000+001, .20000000+001, .20000000+001, .20000000+001, .20000000+001, .20000000+001, .20000000+001,
 .00000000    , .00000000    , .00000000    , .00000000    , .00000000    , .00000000    , .00000000    , .00000000    ,
 .00000000    , .00000000    , .00000000    , .00000000    ,
THRESH = .00000000    , .00000000    , .00000000    , .00000000    , .00000000    , .00000000    , .00000000    , .00000000    ,
 .00000000    , .00000000    , .00000000    , .00000000    , .00000000    , .00000000    , .00000000    , .00000000    ,
 .00000000    , .00000000    , .00000000    , .00000000    ,NCLASS =          8.,IUNIT =         10.,IPROB =          0.
FACTOR = .10000000-004,POWER = .00000000
$END

@ADD,P NORDATA.MA-HOUR20
```

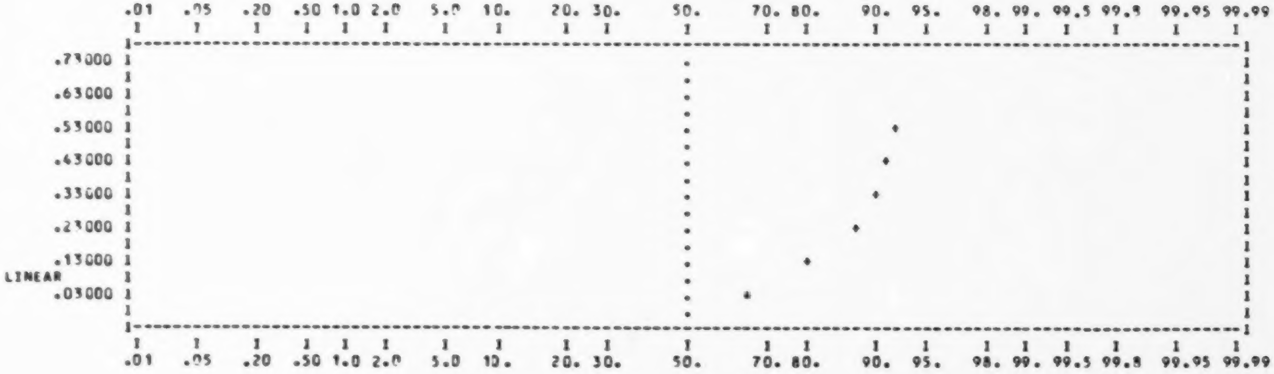
INPUT VARIABLES

SAMPLE	1	2	3	4	5	6	7	8
1	2.70000-003	1.27000-002	3.00000-002	-1.43000-002	4.0000	160.00	3.9000	-8.2000
2	1.64000-002	3.27000-002	1.11000-002	5.28000-002	5.0000	140.00	-1.7000	-8.7000
3	2.53000-002	7.26000-002	2.80000-003	5.86000-002	.00000	.00000	.00000	-4.7000
4	1.04000-002	6.23000-002	1.11000-002	.21310	4.0000	190.00	4.9000	-1.5000
5	-5.00000-004	1.25000-002	2.32000-002	-8.70000-003	9.0000	140.00	-7.3000	-12.400
6	.13120	.17460	-1.20000-003	7.00000-003	11.000	120.00	-4.7000	-9.5000
7	5.25000-002	5.65000-002	2.10000-003	.76540	3.0000	90.000	-3.6000	-9.5000
8	-5.60000-003	-2.10000-003	3.09000-002	-1.09000-002	4.0000	340.00	-2.8000	-10.100
9	3.80000-003	3.70000-002	8.00000-003	1.98000-002	.00000	.00000	-2.7000	-6.2000
10	4.26000-002	9.17000-002	1.70000-003	.18500	3.0000	260.00	-3.3000	-5.6000
11	5.16000-002	.10730	1.20000-003	.27280	.00000	.00000	2.5000	-1.0000
12	9.60000-003	4.07000-002	7.10000-003	-5.40000-003	7.0000	350.00	-3.0000	-5.5000
13	2.40000-002	7.41000-002	6.70000-003	.11110	2.0000	60.000	-2.0000	-5.6000
14	-2.40000-003	2.90000-003	4.33000-002	5.00000-003	18.000	170.00	2.9000	-5.4000
15	6.50000-003	2.33000-002	2.83000-002	8.60000-002	8.0000	160.00	-.80000	-7.0000
16	2.44000-002	7.41000-002	6.60000-003	2.70000-003	2.0000	30.000	-2.0000	-5.6000
17	5.60000-003	3.37000-002	1.40000-002	.48030	.00000	.00000	-1.5000	-5.8000
18	3.60000-002	6.90000-002	6.00000-004	.20780	3.0000	310.00	5.8000	-1.5000
19	-6.90000-003	-5.40000-003	4.84000-002	3.80000-003	20.000	170.00	15.300	-7.6000
20	-6.80000-003	1.40000-003	3.81000-002	2.60000-003	6.0000	140.00	3.3000	-12.000
21	-2.90000-003	5.50000-003	4.75000-002	2.58000-002	4.0000	220.00	6.3000	1.0000
22	1.20000-003	5.20000-003	4.25000-002	-6.60000-003	11.000	210.00	15.000	.80000
23	2.10000-003	5.50000-003	3.61000-002	-6.60000-003	20.000	340.00	11.200	.50000
24	1.60000-003	2.29000-002	2.14000-002	7.70000-003	2.0000	20.000	6.3000	-2.3000
25	-8.00000-004	1.69000-002	3.19000-002	2.00000-004	6.0000	150.00	18.100	2.2000
26	-1.80000-003	1.69000-002	2.17000-002	-1.55000-002	5.0000	250.00	15.000	-.60000
27	1.51000-002	1.68000-002	3.10000-002	1.35000-002	10.000	190.00	24.100	-4.7000
28	-5.40000-003	1.70000-003	4.02000-002	1.81000-002	9.0000	140.00	17.500	-1.4000
29	-6.00000-004	9.20000-003	3.60000-002	1.43000-002	5.0000	50.000	8.6000	-5.3000
30	-7.10000-003	-5.60000-003	3.23000-002	1.28000-002	8.0000	140.00	26.900	-4.6000
31	5.30000-003	1.00000-002	2.65000-002	1.27000-002	10.000	160.00	13.100	1.2000
32	1.12000-002	5.05000-002	3.07000-002	8.80000-003	3.0000	270.00	24.400	6.3000
33	4.20000-003	1.22000-002	4.56000-002	3.40000-003	9.0000	160.00	31.000	.70000
34	6.40000-003	1.49000-002	4.21000-002	1.51000-002	8.0000	350.00	26.000	1.3000
35	-1.00000-004	1.06000-002	4.54000-002	.14020	7.0000	310.00	21.700	-1.5000
36	7.50000-003	1.89000-002	1.72000-002	2.60000-003	5.0000	180.00	19.500	-.40000
37	3.50000-003	1.00000-002	3.54000-002	9.10000-003	18.000	190.00	26.400	-2.0000
38	-3.30000-003	2.32000-002	3.17000-002	7.70000-003	6.0000	340.00	23.200	-6.4000
39	-1.30000-003	1.69000-002	3.04000-002	5.81000-002	8.0000	350.00	19.000	-6.9000
40	-7.00000-004	7.40000-003	3.23000-002	9.60000-003	6.0000	330.00	26.500	-3.6000
41	.00000	1.11000-002	4.25000-002	3.73000-002	6.0000	240.00	14.800	-2.1000
42	5.60000-003	6.00000-003	3.52000-002	.41300	3.0000	20.000	15.300	-2.6000
43	7.00000-004	3.00000-003	3.09000-002	7.60000-003	9.0000	340.00	7.4000	.40000
44	6.60000-003	2.31000-002	2.84000-002	7.51000-002	6.0000	160.00	14.700	-5.6000
45	.10650	.14340	1.08000-002	1.24000-002	7.0000	160.00	18.100	2.3000
46	5.70000-002	9.17000-002	4.00000-003	.25380	4.0000	170.00	6.1000	-3.3000
47	1.32000-002	4.09000-002	1.93000-002	6.88000-002	5.0000	200.00	9.7000	1.0000
48	7.35000-002	8.59000-002	2.49000-002	1.23000-002	5.0000	110.00	17.500	-.80000

```
                                              ANALYSIS OF VARIABLE  4

MEAN VALUE                                          =    .0760
STANDARD DEVIATION                                  =    .1485


CELL NUMBER      FREQUENCY
      1              0.
      2              0.
      3             31.
      4              6.
      5              2.
      6              1.
      7              3.
      8              5.

CHI-SQUARE                                          = 124.6667
PROB. OF EXCEEDING CHI-SQUARE IF NULL HYP. IS TRUE =    .0000
DEGREES OF FREEDOM                                  =    5
```



CUMULATIVE DISTRIBUTION CHART



CUMULATIVE FREQUENCY (PERCENT) - NORMAL PROBABILITY SCALE

```
                                        TRANSFORMED VARIABLES

SAMPLE     1           2           3           4             5           6           7           8

   1      .69804      .70226      .70863      .69375       -1.3681      3.7588      3.9000     -8.2000
   2      .70483      .71212      .69928      .72673       -3.2139      3.8302     -1.7000     -8.7000
   3      .70922      .73150      .69515      .72953  -8.41437-012 4.64260-006-1.02900-005     -4.7000
   4      .70186      .72653      .69928      .80137        .69461      3.9392      4.9000     -1.5000
   5      .69644      .70216      .70527      .69654       -5.7851      6.8944     -7.3000     -12.400
   6      .76001      .77942      .69315      .70433       -9.5263      5.5000     -4.7000     -9.5000
   7      .72251      .72372      .69480      1.0228       -3.0000 5.78169-007     -3.6000     -9.5000
   8      .69390      .69490      .70907      .69544        1.3681     -3.7588     -2.8000     -10.100
   9      .69858      .71422      .69774      .71064  -9.65892-012-4.97410-006     -2.7000     -6.2000
  10      .71769      .74065      .69460      .78868        2.9544      .52094     -3.3000     -5.6000
  11      .72207      .74806      .69435      .82781  -1.36083-011-5.90408-006      2.5000     -.99999
  12      .70146      .71603      .69729      .69619        1.2156     -6.8936     -3.0000     -5.5000
  13      .70858      .73222      .69709      .75452       -1.7320     -1.0000     -2.0000     -5.6000
  14      .69549      .69739      .71515      .70532       -3.1257      17.727      2.9000     -5.4000
  15      .69992      .70749      .70779      .74265       -2.7361      7.5175     -.79999     -7.0000
  16      .70877      .73222      .69804      .70221       -1.0000     -1.7320     -2.0000     -5.6000
  17      .69948      .71261      .70072      .91461  -3.81300-012 3.12524-006     -1.5000     -5.8000
  18      .71447      .72977      .69405      .79899        2.2981     -1.9284      5.8000     -1.5000
  19      .69325      .69325      .71764      .70275       -3.4729      19.696      15.300     -7.6000
  20      .69330      .69664      .71261      .70216       -3.8567      4.5963      3.3000     -12.000
  21      .69524      .69868      .71721      .71359        2.5712      3.0642      6.3000      1.0000
  22      .69729      .69853      .71476      .69759        5.5000      9.5263      15.000      .80000
  23      .69774      .69868      .71163      .69759        6.8404     -18.794      11.200      .50000
  24      .69749      .70730      .70438      .70468       -.68405     -1.8794      6.3000     -2.3000
  25      .69629      .70433      .70956      .70097       -3.0000      5.1962      18.100      2.2000
  26      .69579      .70433      .70453      .69315        4.6985      1.7101      15.000     -.60000
  27      .70419      .70428      .70912      .70754        1.7365      9.8481      24.100     -4.7000
  28      .69400      .69679      .71364      .70981       -5.7851      6.8944      17.500     -1.4000
  29      .69639      .70052      .71158      .70794       -3.8302     -3.2139      8.6000     -5.3000
  30      .69315      .69315      .70976      .70720       -5.1423      6.1284      26.900     -4.6000
  31      .69933      .70092      .70690      .70715       -3.4202      9.3969      13.100      1.2000
  32      .70226      .72081      .70897      .70522        3.0000 3.00405-006      24.400      6.3000
  33      .69878      .70201      .71628      .70255       -3.0782      8.4572      31.000      .70000
  34      .69987      .70334      .71457      .70833        1.3892     -7.8785      26.000      1.3000
  35      .69664      .70121      .71618      .76812        5.3623     -4.4995      21.700     -1.5000
  36      .70042      .70532      .70231      .70216   8.84148-007      5.0000      19.500     -.40000
  37      .69843      .70092      .71128      .70537        3.1257      17.727      26.400     -2.0000
  38      .69505      .70744      .70946      .70468        2.0521     -5.6382      23.200     -6.4000
  39      .69604      .70433      .70882      .72929        1.3892     -7.8785      19.000     -6.9000
  40      .69634      .69963      .70976      .70562        2.5000     -4.3301      26.500     -3.6000
  41      .69669      .70146      .71476      .71920        5.1962      3.0000      14.800     -2.1000
  42      .69948      .69893      .71118      .88727       -1.0261     -2.8191      15.300     -2.6000
  43      .69704      .69744      .70907      .70463        3.0782     -8.4572      7.4000      .39999
  44      .69997      .70740      .70784      .73745       -2.0521      5.6382      14.700     -5.6000
  45      .74839      .76500      .69913      .70700       -2.3941      6.5778      18.100      2.3000
  46      .72469      .74065      .69574      .81947       -.69458      3.9392      6.1000     -3.3000
  47      .70325      .71613      .70334      .73443        1.7101      4.6985      9.7000      1.0000
  48      .73266      .73788      .70611      .70695       -4.6985      1.7101      17.500     -.80000
        THE ORIGINAL RANDOM NUMBER SEED =          1126149989.00
```

6.6.2. NORMIX

The data on output pages 6-129 to 6-134 were selected from the NORMIX output for the MAGNA, Utah, hour 20 data after the four pollutant variates had been transformed by taking natural logarithms. On the first page, note that NTYP stops at 3, indicating that no clusters beyond 3 will be tested. NOR = 1, signifying unequal covariance matrices for clusters. MAXAV = 48, so all 48 sample points go into the hierarchical grouping procedure. Options IHER (for grouping display), MERGE (for final grouping results in plain language), IGEN (for eigenvalues and eigenvectors of the entire data set), IBACK (for back-transformation statistics), and IPROF (for creating a profile-plotting file) are all turned on, i.e., set equal to 1.

Additional settings are for IPOL, identifying the first four variates as variated to be backtransformed, and for NWIND and IWIND, to establish wind conversion to components for the fifth and sixth variates, which are wind speed and direction, respectively. Finally, a table of variate names and units is printed. Page 6-130 gives a more detailed summary of the parameters that begin the second output page.

Skipping several pages of output, page 6-131 shows the means, standard deviations, and correlation coefficient arrays for the eight variates for two clusters, first for the "whole sample" and then for each "type," or cluster. Since "ITERATION = 0" for these statistics, reference is being made to the outcome of the hierarchical grouping and the preliminary computations in the REVALU Subroutine.

Page 6-132 is a subsequent page of output headed by "ITERATION = 101." Since ITERX, the maximum number of iterations, was set at 100, it is evident that convergence to a maximum likelihood solution was not achieved. Therefore the statistics in these tables are questionable. However, the means and standard deviations are reasonable even though the correlations are not. In general, the user can benefit from at least some of the output that follows a convergence failure. The line entitled "BACKTRANSFORMATION TERMS FOR EACH VARIATE ARE" gives the scaling terms that were added to the transformed variates before natural logarithms were taken in NOMRIX-PREP. These terms are used to obtain the cluster statistics in the original units.

The "PROBABILITIES OF TYPE MEMBERSHIP" appearing on page 6-133 depict the outcome of the discriminant analysis in Subroutine ASSIGN. All probabilities are unity in this case, so there is no overlapping between the 8-dimensional clusters. Since the proportion of the population in the second cluster is 0.063, three of the 48 observations are expected in this category. the list of cluster members indicates that the observations are numbers 7, 17, and 42.

The table of backtransformed cluster statistics on page 6-134 is an option that is turned on by setting IBACK = 1. The pollutant variables were backtranformed individually before computation of the means and second moments was carried out. If wind components are to be backtransformed into wind speed and direction, the variates for west component and south component must be adjacent, with IWIND(1) equal to the west component variate number and IWIND(2) equal to the south wind component variate number.

```
@XQT NORTEMP.CLUSTER/CLUSTER
 $LIST
NVPLES =            9.NSAMPL =          48.INP =             0.
NTYP =            1.          2.          3.          0.          0.          0.          0.          0.          0.          0.
             0.          0.          0.          0.          0.          0.          0.          0.          0.          0.
NOR =             1.FNMIN = .00000000000000000    .HTEST = .50000000000000000-001.MAXAV =          48.IPRINT =           0.
LTIME =            2.ITERX =          100.THIER =           0.MERGE =           0.IGEN =           0.MTIME =           0.NTIME =           0.
IHIS =            1.TOVER =            1.IMAP =            1.TBACK =           1.IUNITS =          1.IPROF =           1.
IPOL =            1.          2.          3.          4.          0.          0.          0.          0.          0.          0.
             0.          0.          0.          0.          0.          0.          0.          0.          0.          0.
POWER = .00000000000000000    .NWIND =            1.
IWIND =           5.          6
$END


                              VARIATE NAMES AND UNITS

               1                   NO        NO        PPM
               2                  NOX       NOX        PPM
               3                OZONE        O3        PPM
               4                   TS        TS        PPM
               5           WIND SPEED    W COMP        KPH
               6       WIND DIRECTION    S COMP        KPH
               7          TEMPERATURE         T      DEG C
               8            DEW POINT        DP      DEG C
```

```
                              PROGRAM NORMIX
             WOLFE NORMAL MIXTURE ANALYSIS PROCEDURE(1979 REVISION)

  NORMIX PROGRAM WITH MAGNA DATA, HOUR 20


NVBLES=  8  NSAMPL=  48  DIFF COV MATRIX
MIN CLUSTER SIZE=  9  HYPOTHESIS TEST= .0500
INITIAL KMEANS= 48  TIME LIMIT=  2  ITER LIMIT=100
NO. OF TYPES=  1  2  3  0  0  0  0  0  0  0  0  0  0  0
FORMAT FOR READING DATA=    (6F12.5,/,6F12.5)
END OF FILE REACHED AT LABEL 101
```

```
                                                    PROGRAM NORMIX
                              WOLFE NORMAL MIXTURE ANALYSIS PROCEDURE(1979 REVISION)
NORMIX PROGRAM WITH MAGNA DATA, HOUR 20

      SAMPLE SIZE =          48
      NUMBER OF VARIABLES =  8
      NUMBER OF TYPES =      2

                  ITERATION NUMBER   0

   LIKELIHOOD OF  2 TYPES IN THIS SAMPLE =     .36352437+003

                                        CHARACTERISTICS OF THE WHOLE SAMPLE

                                                         MEANS
    1          2          3          4          5          6          7          8
  .7040      .7127      .7060      .7357     -.3321     2.1207    10.1375    -3.5375

                                                  STANDARD DEVIATIONS
    1          2          3          4          5          6          7          8
  .0138      .0188      .0072      .0644     3.4001     6.9716    10.4548     4.0309

                                                     CORRELATIONS
    1          2          3          4          5          6          7          8
 1.0000      .9319     -.6347      .2429     -.3329      .0311     -.2422     -.0036
  .9319     1.0000     -.7787      .2468     -.2422     -.0440     -.3511      .0083
 -.6347     -.7787     1.0000     -.4042      .1626      .1731      .5990      .2087
  .2429      .2468     -.4042     1.0000     -.0122     -.1249     -.2871     -.1369
 -.3329     -.2422      .1626     -.0122     1.0000     -.4162      .2327      .4220
  .0311     -.0440      .1731     -.1249     -.4162     1.0000      .1009     -.0793
 -.2422     -.3511      .5990     -.2871      .2327      .1009     1.0000      .5720
 -.0036      .0083      .2087     -.1369      .4220     -.0793      .5720     1.0000
             CHARACTERISTICS OF TYPE   1

          THE PROPORTION OF THE POPULATION FROM THIS TYPE=  .938

                                                         MEANS
    1          2          3          4          5          6          7          8
  .7038      .7128      .7063      .7219     -.2648     2.3247    10.5867    -3.3756

                                                  STANDARD DEVIATIONS
    1          2          3          4          5          6          7          8
  .0137      .0188      .0072      .0363     3.3901     6.9267    10.3090     3.9818

                                                     CORRELATIONS
    1          2          3          4          5          6          7          8
 1.0000      .9343     -.6337      .3452     -.3299      .0380     -.2361      .0056
  .9343     1.0000     -.7879      .4573     -.2440     -.0458     -.3583      .0063
 -.6337     -.7879     1.0000     -.5233      .1541      .1602      .5901      .1917
  .3452      .4573     -.5233     1.0000      .0909     -.0559     -.2694     -.0149
 -.3299     -.2440      .1541      .0909     1.0000     -.4289      .2238      .4163
  .0380     -.0458      .1602     -.0559     -.4289     1.0000      .0837     -.0988
 -.2361     -.3583      .5901     -.2694      .2238      .0837     1.0000      .5607
  .0056      .0063      .1917     -.0149      .4163     -.0988      .5607     1.0000
             CHARACTERISTICS OF TYPE   2

          THE PROPORTION OF THE POPULATION FROM THIS TYPE=  .063

                                                         MEANS
    1          2          3          4          5          6          7          8
  .7072      .7118      .7022      .9416    -1.3420     -.9397     3.4000    -5.9657

                                                  STANDARD DEVIATIONS
    1          2          3          4          5          6          7          8
  .0137      .0188      .0072      .0363     3.3901     6.9267    10.3090     3.9818

                                                     CORRELATIONS
    1          2          3          4          5          6          7          8
 1.0000      .9343     -.6337      .3452     -.3299      .0380     -.2361      .0056
  .9343     1.0000     -.7879      .4573     -.2440     -.0458     -.3583      .0063
 -.6337     -.7879     1.0000     -.5233      .1541      .1602      .5901      .1917
  .3452      .4573     -.5233     1.0000      .0909     -.0559     -.2694     -.0149
 -.3299     -.2440      .1541      .0909     1.0000     -.4289      .2238      .4163
  .0380     -.0458      .1602     -.0559     -.4289     1.0000      .0837     -.0988
 -.2361     -.3583      .5901     -.2694      .2238      .0837     1.0000      .5607
  .0056      .0063      .1917     -.0149      .4163     -.0988      .5607     1.0000
```

```
                                                   PROGRAM NORMIX
                              WOLFE NORMAL MIXTURE ANALYSIS PROCEDURE(1979 REVISION)

NORMIX PROGRAM WITH MAGNA DATA, HOUR 20

      SAMPLE SIZE =          48
      NUMBER OF VARIATES =    8
      NUMBER OF TYPES =       2

                    ITERATION NUMBER 101

  LOG LIKELIHOOD OF  2 TYPES IN THIS SAMPLE =     .38982893+003

                                              CHARACTERISTICS OF THE WHOLE SAMPLE

     NO         NOX          O3          TS      W COMP    S COMP          T          DP
    PPM         PPM         PPM         PPM        KPH       KPH      DEG C       DEG C
                                                               MEANS
     1           2           3           4          5         6          7           8
   .7040       .7127       .7C60       .7357     -.3321    2.1207    10.1375     -3.5375

                                                       STANDARD DEVIATIONS
     1           2           3           4          5         6          7           8
   .0138       .0188       .0C72       .0644     3.4001    6.9716    10.4548      4.0309

                                                           CORRELATIONS
     1           2           3           4          5         6          7           8
  1.0000       .9319      -.6347       .2429     -.3329     .0311     -.2422      -.0036
   .9319      1.0000      -.7787       .2468     -.2422    -.0440     -.3511       .0083
  -.6347      -.7787      1.0000      -.4042      .1626     .1731      .5990       .2087
   .2429       .2468      -.4042      1.0000     -.0122    -.1249     -.2871      -.1369
  -.3329      -.2422       .1626      -.0122     1.0000    -.4162      .2327       .4220
   .0311      -.0440       .1731      -.1249     -.4162    1.0000      .1009      -.0793
  -.2422      -.3511       .5990      -.2871      .2327     .1009     1.0000       .5720
  -.0036       .0083       .2087      -.1369      .4220    -.0793      .5720      1.0000

                 CHARACTERISTICS OF TYPE   1

            THE PROPORTION OF THE POPULATION FROM THIS TYPE=  .937

     NO         NOX          O3          TS      W COMP    S COMP          T          DP
    PPM         PPM         PPM         PPM        KPH       KPH      DEG C       DEG C
                                                               MEANS
     1           2           3           4          5         6          7           8
   .7038       .7128       .7C63       .7219     -.2648    2.3247    10.5867     -3.3756

                                                       STANDARD DEVIATIONS
     1           2           3           4          5         6          7           8
   .0139       .0193       .0072       .0343     3.4865    7.1456    10.4207      4.0474

                                                           CORRELATIONS
     1           2           3           4          5         6          7           8
  1.0000       .9390      -.6272       .2976     -.3205     .0340     -.2214       .0380
   .9390      1.0000      -.7856       .4490     -.2397    -.0521     -.3428       .0308
  -.6272      -.7856      1.0000      -.4940      .1476     .1760      .5713       .1576




   .2976       .4490      -.4940      1.0000      .1348    -.0751     -.2334       .0594
  -.3205      -.2397       .1476       .1348     1.0000    -.4304      .2242       .4135
   .0340      -.0521       .1760      -.0751     -.4304    1.0000      .0957      -.0932
  -.2214      -.3428       .5713      -.2334      .2242     .0957     1.0000       .5483
   .0380       .0308       .1576       .0594      .4135    -.0932      .5483      1.0000

                 CHARACTERISTICS OF TYPE   2

            THE PROPORTION OF THE POPULATION FROM THIS TYPE=  .063

     NO         NOX          O3          TS      W COMP    S COMP          T          DP
    PPM         PPM         PPM         PPM        KPH       KPH      DEG C       DEG C
                                                               MEANS
     1           2           3           4          5         6          7           8
   .7072       .7118       .7022       .9416    -1.3420    -.9397     3.4000     -5.9667

                                                       STANDARD DEVIATIONS
     1           2           3           4          5         6          7           8
   .0109       .0101       .0C68       .0585     1.2450    1.9289     8.4581      2.8194

                                                           CORRELATIONS
     1           2           3           4          5         6          7           8
  1.0000       .8346      -.7761       .9816     -.9417     .5000     -.5852      -.8862
   .8346      1.0000      -.9951       .9243     -.6006     .8964     -.9351      -.9948
  -.7761      -.9951      1.0000      -.8822      .5187    -.9341      .9455       .9800
   .9816       .9243      -.8822      1.0000     -.8602     .6560     -.7291      -.9583
  -.9417      -.6006       .5187      -.8602     1.0000    -.1794      .2782       .6786
   .5000       .8964      -.9341       .6560     -.1794    1.0000     -.9948      -.8444
  -.5852      -.9351       .9455      -.7291      .2782    -.9948     1.0000       .6943
  -.8862      -.9948       .9800      -.9583      .6786    -.8444      .8943      1.0000

      BACKTRANSFORMATION TERMS FOR EACH VARIATE ARE
  .20071+001  .20054+001  .20012+001  .20155+001  .00000      .00000      .00000      .00000
```

```
                                        PROGRAM NORMIX
                    WOLFE NORMAL MIXTURE ANALYSIS PROCEDURE(1979 REVISION)

NORMIX PROGRAM WITH MAGNA DATA, HOUR 20

     PROBABILITIES OF TYPE MEMBERSHIP

       1      2

  1 1.000   .000
  2 1.000   .000
  3 1.000   .000
  4 1.000   .000
  5 1.000   .000
  6 1.000   .000
  7   .000 1.000
  8 1.000   .000
  9 1.000   .000
 10 1.000   .000
 11 1.000   .000
 12 1.000   .000
 13 1.000   .000
 14 1.000   .000
 15 1.000   .000
 16 1.000   .000
 17   .000 1.000
 18 1.000   .000
 19 1.000   .000
 20 1.000   .000
 21 1.000   .000
 22 1.000   .000
 23 1.000   .000
 24 1.000   .000
 25 1.000   .000
 26 1.000   .000
 27 1.000   .000
 28 1.000   .000
 29 1.000   .000
 30 1.000   .000
 31 1.000   .000
 32 1.000   .000
 33 1.000   .000
 34 1.000   .000
 35 1.000   .000
 36 1.000   .000
 37 1.000   .000
 38 1.000   .000
 39 1.000   .000
 40 1.000   .000
 41 1.000   .000
 42   .000 1.000
 43 1.000   .000
 44 1.000   .000
 45 1.000   .000
 46 1.000   .000
 47 1.000   .000
 48 1.000   .000
```

```
                                                    PROGRAM NORMIX
                                 WOLFE NORMAL MIXTURE ANALYSIS PROCEDURE(1979 REVISION)

NORMIX PROGRAM WITH MAGNA DATA, HOUR 70

    PROBABILITIES OF TYPE MEMBERSHIP

         1      2

  46 1.000   .000
  47 1.000   .000
  48 1.000   .000


                                         CHARACTERISTICS OF TYPE 1 IN BACKTRANSFORMED VALUES

                                         THE PROPORTION OF THE POPULATION FROM THIS TYPE= .937

     1          2          3          4          5          6          7          8
     NO         NOX        O3         TS         SPEED      DIREC      T          DP
     PPM        PPM        PPM        PPM        KPH        DEG        DEG C      DEG C

                                                            MEANS

   .0145      .0344      .0253      .0442      6.8223   188.2223    10.5867    -3.3756
                                                      STANDARD DEVIATIONS

   .0289      .0404      .0147      .0737      4.7592   102.8302    10.5385     4.0931
                                                 MEAN + 2 STANDARD DEVIATIONS

   .0724      .1152      .0548      .1915     16.3406   393.8827    31.6636     4.8107
                                                 MEAN - 2 STANDARD DEVIATIONS

  -.0433     -.0463     -.0042     -.1032     -2.6961   -17.4381   -10.4903   -11.5618


                                         CHARACTERISTICS OF TYPE 2 IN BACKTRANSFORMED VALUES

                                         THE PROPORTION OF THE POPULATION FROM THIS TYPE= .063

     1          2          3          4          5          6          7          8
     NO         NOX        O3         TS         SPEED      DIREC      T          DP
     PPM        PPM        PPM        PPM        KPH        DEG        DEG C      DEG C

                                                            MEANS

   .0212      .0321      .0171      .5529      2.0000    36.6669     3.4000    -5.9667
                                                      STANDARD DEVIATIONS





   .0271      .0253      .0168      .1871      1.7321    47.2581    10.3591     3.4530
                                                 MEAN + 2 STANDARD DEVIATIONS

   .0754      .0826      .0506      .9272      5.4641   131.1830    24.1181      .9394
                                                 MEAN - 2 STANDARD DEVIATIONS

  -.0329     -.0185     -.0164      .1787     -1.4641   -57.8492   -17.3181   -12.8727
```

6.6.3. PROFILE

A CALCOMP "Profile Plot" is presented for three clusters in figure F-1, and the data file that was stored by setting IPROF = 1 is on page 6-138. This data file is called by the PROFILE Routine.

The chart has reordered variates, an option that is turned on by setting IFLAG = 1. The result of the reordering is that the largest cluster, i.e., the cluster with a proportion of 0.875, rises monotonically across the graph. The two small clusters of three observations each are out of phase for variate 6 (south wind component). One of these clusters has mean values that are extremely high in the cases of variates 1 (NO) and 2 (NO_x). The other cluster has a mean that exceeds the 3-sigma line in the case of variate 4 (TS). Both clusters are low in the case of variate 3 (O_3). Further conclusions can be made regarding the variability of observations within the clusters.

6.6.4 ELLIPSE

CALCOMP plots of ellipses for the Magna data set are displayed in Figure F-2, and the corresponding data file that was stored by setting IMAP = 1, is on pages 6-140 to 6-144. This data file is called by the ELLIPSE routine. The means and standard deviations printed in the box (see Figure F-2a) are simply rounded values equivalent to those of the NORMIX discussion in Section 2 of this appendix.

The 30 percent and 60 percent probability ellipses for TS versus the south wind component in km/h (see Figure F-2b) are free of cluster overlap. The smaller cluster with but three observations represents the three greatest TS values in the set. The wind component is very light and northerly in these cases. On the other hand, the rest of the values in the set are mostly on the low side of the centroid, i.e., the right side of the vertical line, and the values are strung out over a broad range of both south and north wind components.

The 30 percent and 60 percent probability ellipses for TS versus the west wind component in km/h (see Figure F-2c) are also nonoverlapping. The three high TS occurrences are associated with east winds and the rest of the observations show no correlation. Thus, the most outstanding TS cases at 2000 hours at Magna occur with very light easterly and northeasterly winds.

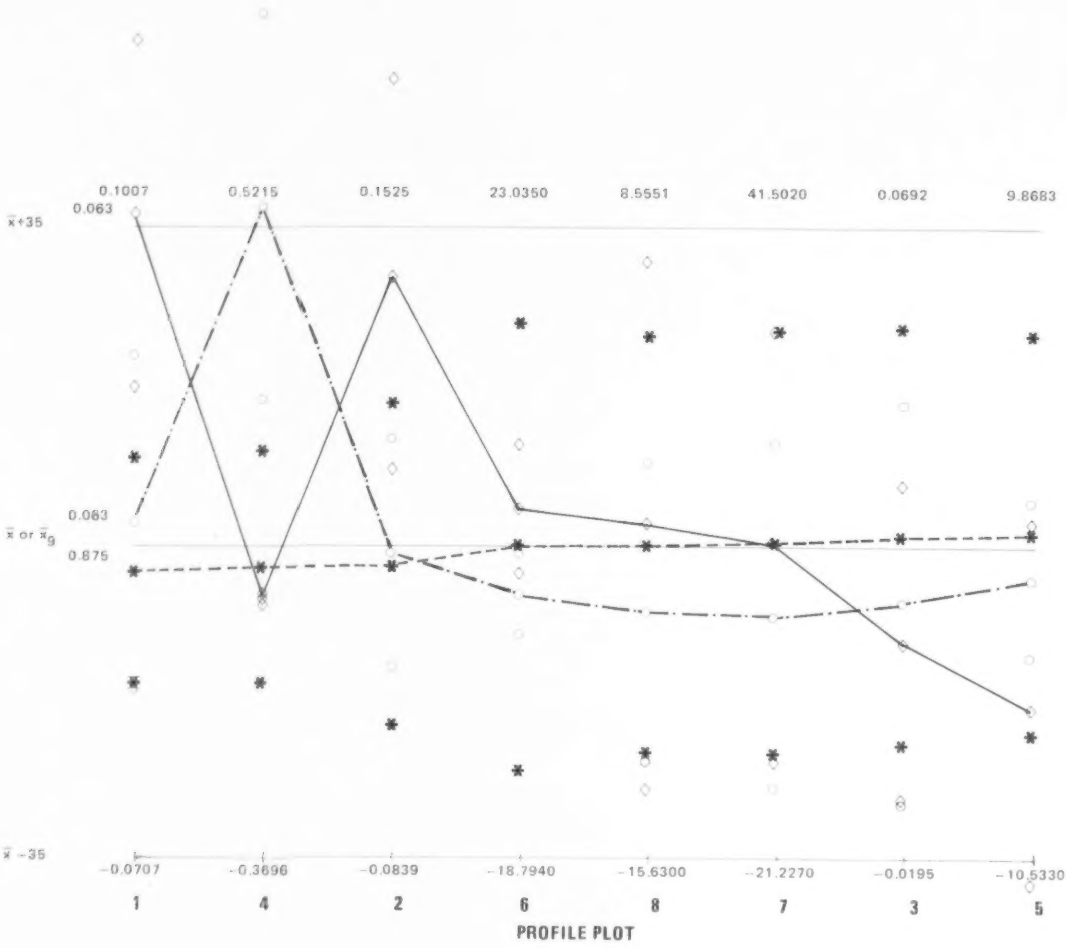
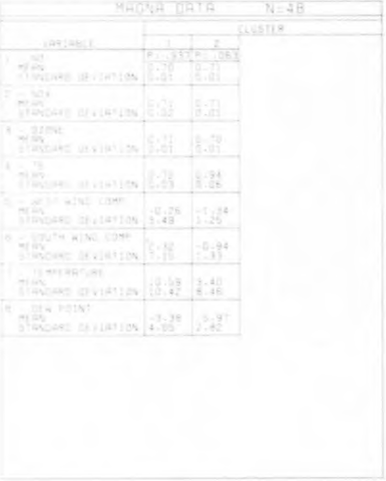


Figure F-1. Cluster variate profile plot for three clusters with eight reordered variates.

```
@DATA,L MAGPRF.
DATA 8R1 73R1Q6 03/21/80 13:06:57 (1)
     1.          8    2
     2.        .937  .063
     3.        .10068+000  .15248+000  .69195-001  .52151+000  .98683+001  .23035+002  .41502+002  .85551+001
     4.       -.70784-001 -.83887-001 -.19614-001 -.36960+000 -.10533+002 -.18794+002 -.21227+002 -.15630+002
     5.                   -.50733-002             .66870+000             -.67885+000
     6.                    .76100-001             .60222+000             -.45002+000
     7.                    .11414-002             .68334+000             -.66106+000
     8.                   -.17121-001             .34195+000             -.37619+000
     9.                    .11667-001             .67455+000             -.65121+000
    10.                   -.17501+000             .44889+000             -.79891+000
    11.                   -.71095-001             .28400+000             -.42619+000
    12.                    .10664+001             .16726+001              .46022+000
    13.                    .66005-002             .69020+000             -.67700+000
    14.                   -.99007-001             .14509+000             -.34311+000
    15.                    .97550-002             .69306+000             -.67355+000
    16.                   -.14632+000            -.19243-001             -.27341+000
    17.                    .14321-001             .67882+000             -.65018+000
    18.                   -.21481+000             .32453+000             -.75416+000
    19.                    .13392-001             .68279+000             -.65601+000
    20.                   -.20088+000             .26542+000             -.66718+000
    21.          8    3
    22.        .875  .063  .063
    23.        .10068+000  .15248+000  .69195-001  .52151+000  .98683+001  .23035+002  .41502+002  .85551+001
    24.       -.70784-001 -.83887-001 -.19614-001 -.36960+000 -.10533+002 -.18794+002 -.21227+002 -.15630+002
    25.                   -.79865-001             .28027+000             -.44000+000
    26.                    .10420+001             .15851+001              .49888+000
    27.                    .76099-001             .60222+000             -.45002+000
    28.                   -.59549-001             .44152+000             -.56062+000
    29.                    .85081+000             .14607+001              .24091+000
    30.                   -.17121-001             .34195+000             -.37619+000
    31.                    .34116-001             .68493+000             -.61670+000
    32.                   -.30261+000             .18531+000             -.79052+000
    33.                   -.17501+000             .44889+000             -.79891+000
    34.                   -.65232-001             .29949+000             -.42996+000
    35.                   -.15317+000            -.14025+000             -.16609+000
    36.                    .10664+001             .16726+001              .46022+000
    37.                    .43538-001             .67169+000             -.58462+000
    38.                   -.51052+000             .72170-001             -.10932+001
    39.                   -.99007-001             .14509+000             -.34311+000
    40.                    .19980-002             .70471+000             -.70071+000
    41.                    .11835+000             .31798+000             -.81270-001
    42.                   -.14632+000            -.19244-001             -.27341+000
    43.                    .14974-001             .67858+000             -.64864+000
    44.                    .51811-002             .68171+000             -.67135+000
    45.                   -.21481+000             .32453+000             -.75416+000
    46.                    .92048-002             .66518+000             -.64677+000
    47.                    .72014-001             .89812+000             -.75410+000
    48.                   -.20088+000             .26542+000             -.66718+000
END DATA.

@FIN
```



MAGNA DATA N=48		
	CLUSTER	
VARIABLE	1	2
	P=.937	P=.063
1 - NO MEAN STANDARD DEVIATION	 0.70 0.01	 0.71 0.01
2 - NOX MEAN STANDARD DEVIATION	 0.71 0.02	 0.71 0.01
3 - OZONE MEAN STANDARD DEVIATION	 0.71 0.01	 0.70 0.01
4 - TS MEAN STANDARD DEVIATION	 0.72 0.03	 0.94 0.06
5 - WEST WIND COMP MEAN STANDARD DEVIATION	 -0.26 3.48	 -1.84 1.25
6 - SOUTH WIND COMP MEAN STANDARD DEVIATION	 2.32 7.15	 -0.94 1.33
7 - TEMPERATURE MEAN STANDARD DEVIATION	 10.59 10.42	 3.40 8.46
8 - DEW POINT MEAN STANDARD DEVIATION	 -3.38 4.05	 -5.97 2.82

Figure F-2a.

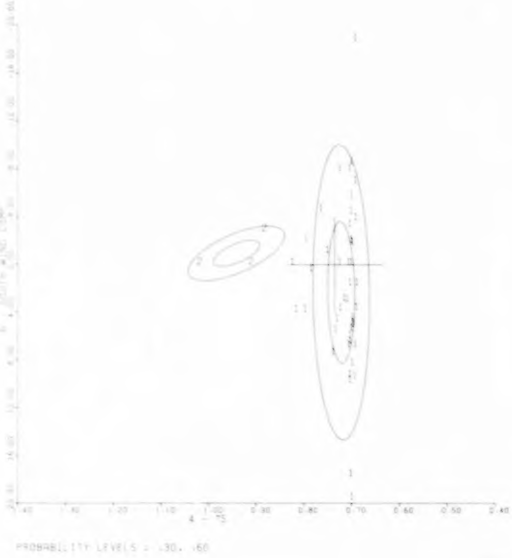


Figure F-2b.

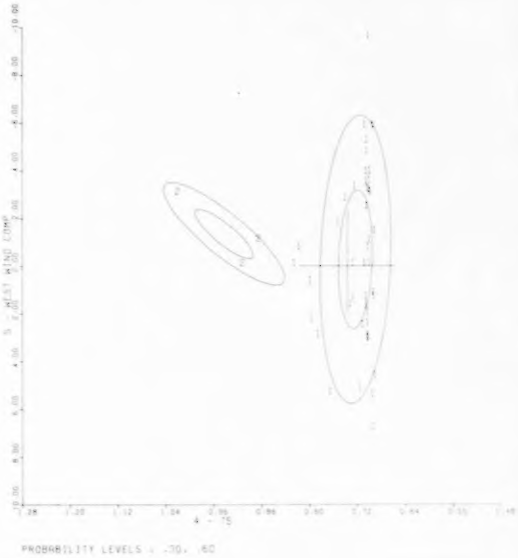


Figure F-2c.

Figure F-2. CALCOMP ellipse plot for two clusters and eight variates.

```
@DATA,L MAGMAP.
DATA 8R1 73RI Q6 03/21/80 13:06:29 (1)
    1.                 8          48
    2.          .69804+000  .70226+000  .70863+000  .69375+000 -.13681+001  .37588+001
    3.          .39000+001 -.82000+001
    4.          .70483+000  .71212+000  .69928+000  .72673+000 -.32139+001  .38302+001
    5.         -.17000+001 -.87000+001
    6.          .70922+000  .73150+000  .69515+000  .72953+000  .00000      .00000
    7.          .00000     -.47000+001
    8.          .70186+000  .72653+000  .69928+000  .80137+000  .69460+000  .39392+001
    9.          .49000+001 -.15000+001
   10.          .69644+000  .70216+000  .70527+000  .69654+000 -.57851+001  .68944+001
   11.         -.73000+001 -.12400+002
   12.          .76001+000  .77942+000  .69315+000  .70433+000 -.95263+001  .55000+001
   13.         -.47000+001 -.95000+001
   14.          .72251+000  .72372+000  .69480+000  .10228+001 -.30000+001  .20087-005
   15.         -.36000+001 -.95000+001
   16.          .69390+000  .69490+000  .70907+000  .69544+000  .13681+001 -.37588+001
   17.         -.28000+001 -.10100+002
   18.          .69858+000  .71422+000  .69774+000  .71064+000  .00000      .00000
   19.         -.27000+001 -.62000+001
   20.          .71769+000  .74065+000  .69460+000  .78868+000  .29544+001  .52094+000
   21.         -.33000+001 -.56000+001
   22.          .72207+000  .74806+000  .69435+000  .82781+000  .00000      .00000
   23.          .25000+001 -.10000+001
   24.          .70146+000  .71603+000  .69729+000  .69619+000  .12155+001 -.68937+001
   25.         -.30000+001 -.55000+001
   26.          .70858+000  .73222+000  .69709+000  .75452+000 -.17321+001 -.10000+001
   27.         -.20000+001 -.56000+001
   28.          .69549+000  .69739+000  .71515+000  .70532+000 -.31256+001  .17727+002
   29.          .29000+001 -.54000+001
   30.          .69992+000  .70749+000  .70779+000  .74265+000 -.27362+001  .75175+001
   31.         -.80000+000 -.70000+001
   32.          .70877+000  .73222+000  .69804+000  .70221+000 -.10000+001 -.17321+001
   33.         -.20000+001 -.56000+001
   34.          .69948+000  .71261+000  .70072+000  .91461+000  .00000      .00000
   35.         -.15000+001 -.58000+001
   36.          .71447+000  .72977+000  .69405+000  .79899+000  .22981+001 -.19284+001
   37.          .58000+001 -.15000+001
   38.          .69325+000  .69325+000  .71764+000  .70275+000 -.34729+001  .19696+002
   39.          .15300+002 -.76000+001
   40.          .69330+000  .69664+000  .71261+000  .70216+000 -.38567+001  .45963+001
   41.          .33000+001 -.12000+002
   42.          .69524+000  .69868+000  .71721+000  .71359+000  .25712+001  .30642+001
   43.          .63000+001  .10000+001
   44.          .69729+000  .69853+000  .71476+000  .69759+000  .55000+001  .95263+001
   45.          .15000+002  .80000+000
   46.          .69774+000  .69868+000  .71163+000  .69759+000  .68404+001 -.18794+002
   47.          .11200+002  .50000+000
   48.          .69749+000  .70730+000  .70438+000  .70468+000 -.68404+000 -.18794+001
   49.          .63000+001 -.23000+001
   50.          .69629+000  .70433+000  .70956+000  .70097+000 -.30000+001  .51962+001
   51.          .18100+002  .22000+001
   52.          .69579+000  .70433+000  .70453+000  .69315+000  .46985+001  .17101+001
   53.          .15000+002 -.60000+000
   54.          .70419+000  .70428+000  .70912+000  .70754+000  .17365+001  .98481+001
   55.          .24100+002 -.47000+001
   56.          .69400+000  .69679+000  .71364+000  .70981+000 -.57851+001  .68944+001
   57.          .17500+002 -.14000+001
   58.          .69639+000  .70052+000  .71158+000  .70794+000 -.38302+001 -.32139+001
   59.          .86000+001 -.53000+001
   60.          .69315+000  .69315+000  .70976+000  .70720+000 -.51423+001  .61284+001
   61.          .26900+002 -.46000+001
   62.          .69933+000  .70092+000  .70690+000  .70715+000 -.34202+001  .93969+001
   63.          .13100+002  .12000+001
   64.          .70226+000  .72081+000  .70897+000  .70522+000  .30000+001 -.59366-005
```

```
 65.        .24400+002  .63000+001
 66.        .69878+000  .70201+000  .71628+000  .70255+000 -.30782+001  .84572+001
 67.        .31000+002  .70000+000
 68.        .69987+000  .70334+000  .71457+000  .70833+000  .13892+001 -.78785+001
 69.        .26000+002  .13000+001
 70.        .69664+000  .70121+000  .71618+000  .76812+000  .53623+001 -.44995+001
 71.        .21700+002 -.15000+001
 72.        .70042+000  .70532+000  .70231+000  .70216+000  .66956-005  .50000+001
 73.        .19500+002 -.40000+000
 74.        .69843+000  .70092+000  .71128+000  .70537+000  .31257+001  .17727+002
 75.        .26400+002 -.20000+001
 76.        .69505+000  .70744+000  .70946+000  .70468+000  .20521+001 -.56382+001
 77.        .23200+002 -.64000+001
 78.        .69604+000  .70433+000  .70882+000  .72929+000  .13892+001 -.78785+001
 79.        .19000+002 -.69000+001
 80.        .69634+000  .69963+000  .70976+000  .70562+000  .25000+001 -.43301+001
 81.        .26500+002 -.36000+001
 82.        .69669+000  .70146+000  .71476+000  .71920+000  .51962+001  .30000+001
 83.        .14800+002 -.21000+001
 84.        .69948+000  .69893+000  .71118+000  .88727+000 -.10261+001 -.28191+001
 85.        .15300+002 -.26000+001
 86.        .69704+000  .69744+000  .70907+000  .70463+000  .30782+001 -.84572+001
 87.        .74000+001  .40000+000
 88.        .69997+000  .70740+000  .70784+000  .73745+000 -.20521+001  .56382+001
 89.        .14700+002 -.56000+001
 90.        .76839+000  .76500+000  .69913+000  .70700+000 -.23941+001  .65779+001
 91.        .18100+002  .23000+001
 92.        .72469+000  .74065+000  .69574+000  .81947+000 -.69659+000  .39392+001
 93.        .61000+001 -.33000+001
 94.        .70325+000  .71613+000  .70334+000  .73443+000  .17101+001  .46985+001
 95.        .97000+001  .10000+001
 96.        .73266+000  .73788+000  .70611+000  .70695+000 -.46985+001  .17101+001
 97.        .17500+002 -.80000+000
 98.        .70402+000  .71272+000  .70603+000  .73567+000 -.33213+000  .21207+001
 99.        .10137+002 -.35375+001
100.        .13757-001  .18824-001  .72364-002  .64357-001  .34001+001  .69716+001
101.        .10455+002  .40309+001
102.          2     1
103.       .937 .063
104.        .70381+000  .71278+000  .70629+000  .72194+000 -.26480+000  .23247+001
105.        .10587+002 -.33756+001
106.        .13903-001  .19262-001  .71953-002  .34279-001  .34865+001  .71456+001
107.        .10421+002  .40474+001
108.        .10000+001  .93904+000 -.62723+000  .29762+000 -.32049+000  .34015-001
109.       -.22139+000  .37957-001
110.        .93904+000  .10000+001 -.78540+000  .44903+000 -.23972+000 -.52080-001
111.       -.34285+000  .30761-001
112.       -.62723+000 -.78540+000  .10000+001 -.49603+000  .14765+000  .17599+000
113.        .57125+000  .15760+000
114.        .29762+000  .44903+000 -.49603+000  .10000+001  .13477+000 -.75078-001
115.       -.23361+000  .59381-001
116.       -.32049+000 -.23972+000  .14765+000  .13477+000  .10000+001 -.43044+000
117.        .22421+000  .41353+000
118.        .34015-001 -.52080-001  .17599+000 -.75078-001 -.43044+000  .10000+001
119.        .95670-001 -.93208-001
120.       -.22139+000 -.34285+000  .57125+000 -.23361+000  .22421+000  .95670-001
121.        .10000+001  .54833+000
122.        .37957-001  .30761-001  .15760+000  .59381-001  .41353+000 -.93208-001
123.        .54833+000  .10000+001
124.        .70716+000  .71175+000  .70223+000  .94156+000 -.13420+001 -.93970+000
125.        .34000+001 -.59667+001
126.        .10856-001  .10139-001  .67722-002  .58520-001  .12450+001  .13289+001
127.        .84581+001  .28194+001
128.        .10000+001  .83460+000 -.77614+000  .98164+000 -.94169+000  .50000+000
129.       -.58520+000 -.88617+000
130.        .83460+000  .10000+001 -.99511+000  .92434+000 -.60059+000  .89435+000
131.       -.93509+000 -.99484+000
```

```
132.      -.77614+000 -.99511+000  .10000+001 -.88216+000  .51871+000 -.93415+000
133.       .96551+000  .97997+000
134.       .98164+000  .92434+000 -.88216+000  .10000+001 -.86022+000  .65599+000
135.      -.72912+000 -.95828+000
136.      -.94169+000 -.60059+000  .51871+000 -.86022+000  .10000+001 -.17944+000
137.       .27823+000  .67858+000
138.       .50000+000  .89435+000 -.93415+000  .65599+000 -.17944+000  .10000+001
139.      -.99484+000 -.84436+000
140.      -.58520+000 -.93509+000  .96551+000 -.72912+000  .27823+000 -.99484+000
141.       .10000+001  .89432+000
142.      -.88617+000 -.99484+000  .97997+000 -.95828+000  .67858+000 -.84436+000
143.       .89432+000  .10000+001
144.                1           1
145.                2           1
146.                3           1
147.                4           1
148.                5           1
149.                6           1
150.                7           2
151.                8           1
152.                9           1
153.               10           1
154.               11           1
155.               12           1
156.               13           1
157.               14           1
158.               15           1
159.               16           1
160.               17           2
161.               18           1
162.               19           1
163.               20           1
164.               21           1
165.               22           1
166.               23           1
167.               24           1
168.               25           1
169.               26           1
170.               27           1
171.               28           1
172.               29           1
173.               30           1
174.               31           1
175.               32           1
176.               33           1
177.               34           1
178.               35           1
179.               36           1
180.               37           1
181.               38           1
182.               39           1
183.               40           1
184.               41           1
185.               42           2
186.               43           1
187.               44           1
188.               45           1
189.               46           1
190.               47           1
191.               48           1
192.       .70402+000  .71272+000  .70603+000  .73567+000 -.33213+000  .21207+001
193.       .10137+002 -.35375+001
194.       .13757-001  .18824-001  .72364-002  .64357-001  .34001+001  .69716+001
195.       .10455+002  .40309+001
196.          3    1
197.      .875 .063 .063
```

```
198.        .70072+000  .70936+000  .70677+000  .72307+000  .11198+000  .21624+001
199.        .10607+002 -.34262+001
200.        .74314-002  .14148-001  .70643-002  .35209-001  .32037+001  .73465+001
201.        .10407+002  .39662+001
202.        .10000+001  .92291+000 -.77013+000  .78422+000  .93150-001 -.95621-001
203.       -.34359+000  .85964-001
204.        .92291+000  .10000+001 -.85164+000  .75685+000  .88958-001 -.17500+000
205.       -.42709+000  .62690-001
206.       -.77013+000 -.85164+000  .10000+001 -.55975+000  .22396-001  .22131+000
207.        .57688+000  .14292+000
208.        .78422+000  .75685+000 -.55975+000  .10000+001  .94045-001 -.65299-001
209.       -.24733+000  .66271-001
210.        .93150-001  .88958-001  .22396-001  .94045-001  .10000+001 -.44930+000
211.        .19376+000  .43131+000
212.       -.95621-001 -.17500+000  .22131+000 -.65299-001 -.44930+000  .10000+001
213.        .10634+000 -.10192+000
214.       -.34359+000 -.42709+000  .57688+000 -.24733+000  .19376+000  .10634+000
215.        .10000+001  .51148+000
216.        .85964-001  .62690-001  .14292+000  .66271-001  .43131+000 -.10192+000
217.        .51148+000  .10000+001
218.        .74702+000  .76077+000  .69946+000  .70609+000 -.55396+001  .45960+001
219.        .10300+002 -.26667+001
220.        .11208-001  .17221-001  .52961-002  .12470-002  .29718+001  .20875+001
221.        .10609+002  .49949+001
222.        .10000+001  .99612+000 -.99912+000 -.81008+000 -.59603+000  .79642+000
223.       -.80612+000 -.64764+000
224.        .99612+000  .10000+001 -.99154+000 -.75530+000 -.52301+000  .84658+000
225.       -.75088+000 -.57804+000
226.       -.99912+000 -.99154+000  .10000+001  .83399+000  .62924+000 -.77031+000
227.        .83027+000  .67908+000
228.       -.81008+000 -.75530+000  .83399+000  .10000+001  .95363+000 -.29058+000
229.        .99998+000  .97139+000
230.       -.59603+000 -.52301+000  .62924+000  .95363+000  .10000+001  .10894-001
231.        .95563+000  .99763+000
232.        .79642+000  .84658+000 -.77031+000 -.29058+000  .10894-001  .10000+001
233.       -.28415+000 -.55014-001
234.       -.80612+000 -.75088+000  .83027+000  .99998+000  .95563+000 -.28415+000
235.        .10000+001  .97296+000
236.       -.64764+000 -.57804+000  .67908+000  .97139+000  .99763+000 -.55014-001
237.        .97296+000  .10000+001
238.        .70716+000  .71175+000  .70223+000  .94156+000 -.13420+001 -.93970+000
239.        .34000+001 -.59667+001
240.        .10856-001  .10139-001  .63722-002  .58520-001  .12450+001  .13289+001
241.        .84581+001  .28194+001
242.        .10000+001  .83460+000 -.77614+000  .98164+000 -.94169+000  .50000+000
243.       -.58521+000 -.88617+000
244.        .83460+000  .10000+001 -.99511+000  .92435+000 -.60059+000  .89435+000
245.       -.93509+000 -.99484+000
246.       -.77614+000 -.99511+000  .10000+001 -.88216+000  .51871+000 -.93415+000
247.        .96551+000  .97997+000
248.        .98164+000  .92435+000 -.88216+000  .10000+001 -.86023+000  .65600+000
249.       -.72912+000 -.95828+000
250.       -.94169+000 -.60059+000  .51871+000 -.86023+000  .10000+001 -.17944+000
251.        .27823+000  .67858+000
252.        .50000+000  .89435+000 -.93415+000  .65600+000 -.17944+000  .10000+001
253.       -.99485+000 -.84437+000
254.       -.58521+000 -.93509+000  .96551+000 -.72912+000  .27823+000 -.99485+000
255.        .10000+001  .89432+000
256.       -.88617+000 -.99484+000  .97997+000 -.95828+000  .67858+000 -.84437+000
257.        .89432+000  .10000+001
258.                 1           1
259.                 2           1
260.                 3           1
261.                 4           1
262.                 5           1
263.                 6           2
264.                 7           3
```

```
                 8         1
                 9         1
                10         1
                11         3
                12         1
                13         1
                14         2
                15         1
                16         1
                17         3
                18         1
                19         1
                20         2
                21         3
                22         3
                23         1
                24         1
                25         1
                26         3
                27         2
                28         1
                29         1
                30         1
                31         2
                32         1
                33         1
                34         1
                35         1
                36         2
                37         2
                38         1
                39         1
                40         1
                41         3
                42         3
                43         1
                44         2
                45         2
                46         1
                47         3
                48         2


END DATA.

@FIN
```